Philipp Cimiano
Anette Frank
Michael Kohlhase
Benno Stein (Eds.)

LNAI 14638

Robust Argumentation Machines

First International Conference, RATIO 2024
Bielefeld, Germany, June 5–7, 2024
Proceedings

RATIO

Springer

The series Lecture Notes in Artificial Intelligence (LNAI) was established in 1988 as a topical subseries of LNCS devoted to artificial intelligence.

The series publishes state-of-the-art research results at a high level. As with the LNCS mother series, the mission of the series is to serve the international R & D community by providing an invaluable service, mainly focused on the publication of conference and workshop proceedings and postproceedings.

Philipp Cimiano · Anette Frank ·
Michael Kohlhase · Benno Stein
Editors

Robust Argumentation Machines

First International Conference, RATIO 2024
Bielefeld, Germany, June 5–7, 2024
Proceedings

Springer

Editors
Philipp Cimiano
Bielefeld University
Bielefeld, Germany

Anette Frank
Heidelberg University
Heidelberg, Germany

Michael Kohlhase
Friedrich-Alexander-Universität
Erlangen-Nürnberg
Erlangen, Germany

Benno Stein
Bauhaus-Universität Weimar
Weimar, Germany

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Artificial Intelligence
ISBN 978-3-031-63535-9 ISBN 978-3-031-63536-6 (eBook)
https://doi.org/10.1007/978-3-031-63536-6

LNCS Sublibrary: SL7 – Artificial Intelligence

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Cultivating debate and argumentation as a means of finding consensus and solutions that are acceptable compromises for many seems essential, in particular in times of perceived crises and public division. As public debates are to a large extent carried out online, they are often unmanageable, difficult to trace, and difficult to oversee. Understanding the key positions of diverse stakeholders, their key points or arguments, and how they are justified is key to identifying points and opportunities for compromises.

This is where computational argumentation analysis comes in, providing methods to aid the automatic retrieval, analysis, summarization, ranking, and assessment of arguments. The field of argumentation mining and analysis is relatively young. The 1st Workshop on Argument Mining took place ten years ago, in June 2014 in Baltimore, collocated with the 52nd Annual Meeting of the Association for Computational Linguistics. Since this 1st edition of the Argument Mining Workshop, we have witnessed significant advances in the development of approaches that support the automated analysis, summarization, aggregation, retrieval, and ranking of arguments exchanged "in the wild" at large scale. By "in the wild", we mean arguments that are exchanged on the World Wide Web, in discussion portals or other online formats in which users share opinions and viewpoints on topics that are relevant to them. The methods of argumentation analysis have now reached a level of maturity and robustness that makes them applicable to the analysis of real online debates. They enable systems to identify the most important arguments exchanged, summarize and group arguments, or even automatically generate arguments to present different viewpoints and perspectives.

This volume presents recent advances in the development of robust argumentation machines, i.e. systems capable of systematically, efficiently, and adequately summarizing public debates in terms of arguments, positions, key points, stakeholder groups, tracing them back to groups, etc.

The contributions to this volume can be grouped into six areas: (I) Argument Mining, (II) Persuasion and Deliberation, (III) Argument Acquisition, Annotation, and Quality Assessment, (IV) Computational Models of Argumentation, (V) Interactive Argumentation, Recommendation, and Personalization, and (VI) Argument Search and Retrieval. In the following we summarize the contributions of the papers in this volume.

I Argument Mining

The main challenge of Argument Mining is how to identify, extract, and formalize arguments that are exchanged by key stakeholders and actors. The approaches described in this volume consider a wide spectrum of genres ranging from argumentation in social media through to argumentation in scientific texts. An interesting and novel method consists in the application of sequential pattern mining to identify argumentation schemes.

In their paper **“Natural Language Hypotheses in Scientific Papers and How to Tame Them: Suggested Steps for Formalizing Complex Scientific Claims”**, Heger et al. are concerned with the formalization of hypotheses as key elements of argumentation in scientific texts. Specifically, they develop a framework for formalizing hypotheses in the research field of invasion biology. According to their framework, hypotheses consist of three essential elements: a subject, an object, and a hypothesized relationship between them. The framework not only facilitates argumentation analysis, but also helps to convert scientific publications into a machine-readable format.

In their paper **“Weakly Supervised Claim Localization in Scientific Abstracts”**, Brinner et al. present a weak supervision approach that requires only abstract-level supervision to identify and localize arguments in scientific texts. Their approach uses information about the general presence of a claim in a given abstract to extract sections of text that indicate that specific claim. The method is evaluated on the SciFact claim verification and INAS datasets, showing that significant performance in the claim localization task can be achieved without any explicit supervision.

In their paper “**Argument Mining of Attack and Support Patterns in Dialogical Conversations with Sequential Pattern Mining”**, Ruckdeschel et al. apply Sequential Pattern Mining – a common method for finding patterns in large databases – to identify how typical argumentation schemes in user debates develop over time. They investigate a German Twitter corpus on nuclear energy that they divide into different time slices. When applied to the time slices, the approach reveals distinct patterns of support and attack relations between pro and contra arguments in conversational threads. The proposed method can thus be used to analyze diachronic changes in patterns and show how discourses on certain topics can evolve over time.

Following a related approach, in their paper **“Cluster-Specific Rule Mining for Argumentation-Based Classification”**, Klein et al. present a method that combines machine learning with computational models for (structured) argumentation. In this approach, the data set is clustered and then a rule-learning algorithm is used to extract frequent patterns and rules from the resulting clusters. Experiments show that the method significantly improves the baseline approach.

II Debate Analysis and Deliberation

By developing methods for analyzing political discourse and debates, argument mining also plays a central role in developing methods that can support the analysis of political discourse and opinions on important current issues in order to promote deliberation. The present volume features a number of contributions along these lines.

In their paper **“Automatic Analysis of Political Debates and Manifestos: Successes and Challenges”**, Ceron et al. note that political actors typically communicate via different channels: While the parties communicate their core ideas via published manifestos, individual players use the media to express themselves on a daily basis. On the one hand, manifestos are useful to characterize the positions of parties at a global ideological level over time. On the other hand, individual statements can be collected to analyze debates in specific policy areas at a fine-grained level, in terms of individual actors and demands. The authors suggest using NLP-based analysis for these two different channels to highlight the advantages and challenges of both approaches.

In their paper **“PAKT: Perspectivized Argumentation Knowledge Graph and Tool for Deliberation Analysis”**, Plenz et al. present PAKT, a model for deliberation analysis at a structural level that leverages argumentation mining and knowledge graph construction methods. Beyond individual arguments, PAKT uncovers structural patterns in the way participants argue and shows how to characterize the argumentative perspectives of different stakeholder groups using frames, values, and conceptual analysis. In several case studies, the authors show how their perspective argumentation analysis can identify key points for initiating deliberative solutions to facilitate constructive discourse and informed decision-making.

In their paper **“PolArg: Unsupervised Polarity Prediction of Arguments in Real-Time Online Conversations”**, Lenz and Bergmann point out that conversations in social networks often involve numerous participants and take place at a fast pace. They conclude from this that real-time analysis is an important prerequisite for systems for analyzing online conversations. They propose to address this issue using Large Language Models and investigate unsupervised prompting strategies for detecting argumentation polarity in datasets from Kialo, X/Twitter, and Hacker News. The authors show that their approach is more effective for X posts than a model tuned to Kialo debates, and less effective for Hacker News posts, which are less argumentative.

III Argument Acquisition, Annotation, and Quality Assessment

An important topic within argument mining is to evaluate the quality of arguments. This includes the development of models that can automatically predict the quality of arguments.

Mirzakhmedova et al. explore this question in their paper **“Are Large Language Models Reliable Argument Quality Annotators?”**, where they focus in particular on the question of how to reliably annotate arguments for quality. The authors note that due to the high subjectivity involved in the annotation of argument quality, there is often high disagreement and thus inconsistency between human annotators. In this context, the authors investigate the potential of using state-of-the-art large language models as proxies for argument quality annotators. Analyzing the agreement between human experts and novice annotators in comparison to the LLM-based annotations, the authors show that LLMs can produce consistent annotations, with a moderately high agreement with human experts across most of the quality dimensions. Moreover, they show that using LLMs as additional annotators can significantly improve the agreement between annotators.

Continuing the topic of evaluating the quality of arguments, Knaebel et al., in their paper entitled **“The Impact of Argument Arrangement on Essay Scoring”**, investigate whether the quality of student essays can be algorithmically predicted. To this end, they propose a model that aims to capture the “flows” of semantic types of argumentative units. The authors train linear classification models on flow features and find that flows based on semantic types are better predictors of essay quality compared to flows of coarse argument components.

In their paper **"Finding Argument Fragments on Social Media with Corpus Queries and LLMs"**, Dykes et al. address the challenge of compiling a gold standard of high precision for argumentative fragments. To circumvent the need for manual annotation, they present a pattern-based approach that queries a corpus of patterns to extract argumentative fragments. They apply their approach to a large corpus of English tweets on the subject of the UK Brexit referendum in 2016. The authors show how queries can be combined to extract complex nested statements that are relevant for a given argument. The approach further allows adjustment of the trade-off between precision and recall, by setting a cutoff threshold to match the needs of specific applications.

IV Computational Models of Argumentation

In addition to identifying and extracting arguments, an important aspect of a robust argumentation analysis is to evaluate the identified arguments, e.g. to determine the most relevant, strongest, or best arguments in a debate. For this purpose, formal computational models are needed to represent and formalize arguments so that we can reason with them. The papers in this volume take different approaches. On the one hand, they follow the paradigm of abstract argumentation where sets of arguments are encoded as graphs consisting of arguments as nodes and edges representing attack relations. Others follow assumption-based reasoning (ABA) as well as Pearl's probabilistic causal model or Bayesian networks and present clear scientific and methodological advances for each of these paradigms.

The paper **"Enhancing Abstract Argumentation Solvers with Machine Learning-Guided Heuristics: A Feasibility Study"**, by Hoffmann et al. is located in the paradigm of abstract argumentation and focuses on the determination of admissible sets, i.e. sets of arguments that can defend themselves against (external) attacks. The determination of such admissible sets, which depend on a certain semantics, is known to be an NP-hard problem. Building on recent research demonstrating the efficacy of using machine learning to provide approximative solutions, the authors propose a new approach that leverages a random forest classifier to predict acceptability, and subsequently use the predictions to form a heuristic that guides a search-based solver.

The work of Skiba et al. **"Ranking Transition-based Medical Recommendations using Assumption-based Argumentation"** builds on the Assumption-Based Argumentation (ABA) framework and introduces as a new contribution an approach to categorizing assumptions that relies on their relationship to other assumptions and the syntactic structure of the ABA framework. The authors propose a new family of semantics for ABA frameworks that rely on reductions to the abstract argumentation setting and utilize existing rank-based semantics for abstract argumentation. The suitability of the approach is shown in a case study for generating recommendations for patients with multiple health conditions.

In their paper **"Argumentation-based Probabilistic Causal Reasoning"**, Bengel et al. propose a reformulation of Pearl's causal models for probabilistic causal and counterfactual reasoning in terms of an argumentation-based framework: Causal statements are interpreted as arguments in an abstract argumentation framework and the attack relation represents contradicting causal inferences, allowing the reasoning process to be questioned during a query. The framework can then be used to generate argumentative explanations for the (non-)acceptance of the causal statement.

The starting point of the paper **"From Networks to Narratives: Bayes Nets and the Problems of Argumentation"** by Keshmirian et al. is the observation of a tight conceptual connection between the argumentative structure of a problem and its representation as a Bayesian Belief Network (BBN). The primary challenge addressed by the authors is the representation of an argumentative structure that renders the BBN inference transparent to non-experts. In particular, the authors investigate how argument structures can be extracted from BBNs. They show why existing algorithmic approaches to extracting arguments still fall short when it comes to elucidating intricate features of BBNs, such as "explaining away" or other complex interactions between variables. Building on this analysis they suggest future developments to improve the representation of the extracted arguments.

In the paper **"Enhancing Argument Generation Using Bayesian Networks"**, Cao et al. examine algorithms that utilize factor graphs from Bayesian Belief Networks to generate and evaluate arguments. Based on an assessment of the strengths and weaknesses of existing algorithms, they propose an improved algorithm that addresses the identified issues. The proposed algorithmic improvements yield an improved performance on the creation of more robust arguments.

The paper **"Do not disturb my circles!" Identifying the Type of Counterfactual at Hand** by Willig et al. explores the use of causal models to derive explanations. The minimal explanation is a causal chain that does not need any intervention. Possible interventions can be counterfactual interventions, which presuppose intentional interventions, and retrospective counterfactual interventions, which attribute changes to external factors. The approach can decide whether and which measures are required.

V Interactive Argumentation, Recommendation, and Personalization

An important question addressed also in this volume is how users will effectively be able to interact with argumentative systems. In this respect, the paper **"BEA: Building Engaging Argumentation"** by Aicher et al. presents the cooperative argumentative dialog system BEA, which aims to involve the user in a critical discussion of arguments presented. BEA aims to engage users in an intuitive and unbiased opinion formation process, where information can be explored intuitively. Through a virtual agent, BEA can maintain deliberative dialogues with humans. BEA shows how to help users increase their engagement in reflection and conversation.

The paper **"Deciphering Personal Argument Styles - A Comprehensive Approach to Analyzing Linguistic Properties of Argument Preferences"** by Zymla et al. presents an application for exploring the effect of linguistic features on personalized argument preferences. The individual preferences are derived by measuring the impact of linguistic features on pairwise comparisons between arguments. The authors develop a visual interactive labeling system that structures the annotation process of pairwise comparisons. Through these annotations, patterns of argument preferences based on linguistic feature vectors are extracted. By training individual models for different users, the authors show how information can be gained that allows one to compare different user groups and to identify different argumentation preferences across groups.

VI Argument Search and Retrieval

Since the emergence of the first argument search engines such as args.me, the topic of how to support users in finding relevant arguments has received increasing attention in the field of argument mining.

The paper **"Extending the Comparative Argumentative Machine: Multilingualism and Stance Detection"** by Nikishina et al. advances the state of the art in multilingual argument retrieval, focusing on the use case of comparative search, i.e. finding statements that are for or against a set of specific options to be compared. The authors describe how the CAM (comparative argumentative machine) system has been equipped with better answer stance detection capabilities and with system variants to support non-English requests. In order to turn the system into a multilingual system, the authors compare two approaches to support Russian requests and answers: (1) translating the original English CAM data and (2) using an existing replica of CAM on native Russian data. The comparison of the translation-based and replica-based CAM variants in a user study shows that the combination of their responses appears to be the most promising.

The paper **"Objective Argument Summarization in Search"** by Ziegenbein et al. addresses the problem that arguments retrieved from the Web can be of low quality, potentially being long and unstructured, subjective and emotional, and containing inappropriate language. Building on the hypothesis that "objective snippets" of arguments are better suited to be displayed in search results than the commonly used extractive snippets, they develop corresponding methods for two important tasks: snippet generation and neutralization. For these tasks, two approaches are experimentally examined: (1) prompt engineering for large language models (LLMs), and (2) supervised models trained on existing datasets. The authors find that a supervised summarization model outperforms zero-shot summarization with LLMs for snippet generation.

In the paper **"ArgServices: A Microservice-Based Architecture for Argumentation Machines"**, Lenz et al. present a microservices-based architecture for argumentation machines that provides services. The starting point is the fact that the development of argumentation machines is hindered by the lack of common standards and appropriate tools, leading to ad hoc solutions with little reuse value. The proposed architecture provides strongly typed interfaces for the following services: (1) Argument Mining, (2) Case-Based Reasoning on Arguments, (3) Argument Retrieval and Ranking, and (4) Quality Assessment of Arguments. The system has been designed to be extensible, allowing for easy integration of new tasks.

Acknowledgements. The papers collected in this volume collectively represent significant advances in a broad range of challenges and topics in the field of argument mining. Twenty full papers and one short paper were accepted, from twenty-four submissions. Submissions received an average of three single-blind reviews. Many of the contributions included in this volume are a result of the Priority Program "Robust Argumentation

Machines” (RATIO), which was funded by the German Research Foundation (DFG). The editors of this volume would like to thank the DFG for its support.

May 2024

Philipp Cimiano
Anette Frank
Michael Kohlhase
Benno Stein

Organization

General Chair

Philipp Cimiano	Bielefeld University, Germany

Program Committee Chairs

Philipp Cimiano	Bielefeld University, Germany
Anette Frank	Heidelberg University, Germany
Michael Kohlhase	Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
Benno Stein	Bauhaus-Universität Weimar, Germany

Program Committee

Khalid Alkhatib	University of Groningen, Netherlands
Elisabeth Andre	Augsburg University, Germany
Ralph Bergmann	Trier University, Germany
Chris Biemann	Universität Hamburg, Germany
Alexander Bondarenko	Friedrich-Schiller-Universität Jena, Germany
Miriam Butt	University of Konstanz, Germany
Elena Cabrio	Université Côte d'Azur, CNRS, Inria, I3S, France
Philipp Cimiano	Bielefeld University, Germany
Stephanie Evert	Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
Anette Frank	Heidelberg University, Germany
Ivan Habernal	Technical University of Darmstadt, Germany
Matthias Hagen	Friedrich-Schiller-Universität Jena, Germany
Stephan Hartmann	Ludwig Maximilian University of Munich, Germany
Sebastian Haunss	University of Bremen, Germany
Gerhard Heyer	Leipzig University, Germany
Yufang Hou	Technical University of Darmstadt, Germany
Daniel Keim	University of Konstanz, Germany
Michael Kohlhase	Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany

Gabriella Lapesa	University of Stuttgart, Germany
Wolfgang Minker	University of Ulm, Germany
Juri Opitz	Heidelberg University, Germany
Sebastian Pado	University of Stuttgart, Germany
Martin Potthast	Leipzig University, Germany
David Restrepo Amariles	HEC Paris, France
Sebastian Rudolph	Technical University of Dresden, Germany
Ralf Schenkel	Trier University, Germany
Lutz Schröder	Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
Manfred Stede	Universität Potsdam, Germany
Benno Stein	Bauhaus-Universität Weimar, Germany
Matthias Thimm	FernUniversität in Hagen, Germany
Francesca Toni	Imperial College London, UK
Henning Wachsmuth	Leibniz University Hanover, Germany
Sina Zarrieß	Bielefeld University, Germany
Jürgen Ziegler	University of Duisburg-Essen, Germany

Proceedings Chair

Olivia Sánchez-Graillet	Bielefeld University, Germany

Website Administration

Jan-Philipp Töberg	Bielefeld University, Germany

Publicity Chair

Moritz Blum	Bielefeld University, Germany

Contents

Argument Mining

Debate Analysis and Deliberation

Argument Acquisition, Annotation and Quality Assessment

Computational Models of Argumentation

Interactive Argumentation, Recommendation and Personalization

Argument Search and Retrieval

Argument Mining

Natural Language Hypotheses in Scientific Papers and How to Tame Them

Suggested Steps for Formalizing Complex Scientific Claims

Tina Heger[1,2,3](✉), Alsayed Algergawy[4,5], Marc Brinner[6], Jonathan M. Jeschke[1,2], Birgitta König-Ries[4], Daniel Mietchen[1,2,7,8], and Sina Zarrieß[6]

[1] Leibniz Institute of Freshwater Ecology and Inland Fisheries (IGB), Berlin, Germany
t.heger@tum.de
[2] Institute of Biology, Freie Universität Berlin, Berlin, Germany
[3] TUM School of Life Sciences, Technical University of Munich, Freising, Germany
[4] Institute for Informatics, Friedrich-Schiller-University Jena, Jena, Germany
[5] Data and Knowledge Engineering, University of Passau, Passau, Germany
[6] Faculty of Linguistics and Literature Studies, University of Bielefeld, Bielefeld, Germany
[7] Ronin Institute for Independent Scholarship, Montclair, NJ, USA
[8] Institute for Globally Distributed Open Research and Education (IGDORE), Jena, Germany

Abstract. Hypotheses are critical components of scientific argumentation. Knowing established hypotheses is often a prerequisite for following and contributing to scientific arguments in a research field. In scientific publications, hypotheses are usually presented for specific empirical settings, whereas the related general claim is assumed to be known. Prerequisites for developing argumentation machines for assisting scientific workflows are to account for domain-specific concepts needed to understand established hypotheses, to clarify the relationships between specific hypotheses and general claims, and to take steps towards formalization. Here, we develop a framework for formalizing hypotheses in the research field of invasion biology. We suggest conceiving hypotheses as consisting of three basic building blocks: a subject, an object, and a hypothesized relationship between them. We show how the subject-object-relation pattern can be applied to well-known hypotheses in invasion biology and demonstrate that the contained concepts are quite diverse, mirroring the complexity of the research field. We suggest a stepwise approach for modeling them to be machine-understandable using semantic web ontologies. We use the SuperPattern Ontology to categorize hypothesized relationships. Further, we recommend treating every hypothesis as part of a hierarchical system with 'parents' and 'children'. There are three ways of moving from a higher to a lower level in the hierarchy: (i) specification, (ii) decomposition, and (iii) operationalization. Specification involves exchanging subjects or objects. Decomposition means zooming in and making explicit assumptions about underlying (causal) relationships. Finally, operationalizing a hypothesis means providing concrete descriptions of what will be empirically tested.

Keywords: Complex claims · invasion biology · ontology · scientific hypotheses

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 3–19, 2024.
https://doi.org/10.1007/978-3-031-63536-6_1

1 Introduction: Scientific Hypotheses as Complex Claims

In scientific contexts, argumentation is part of established workflows. In an idealized setting, a research question arises from some applied context or a scientific debate. Based on this question, the researcher formulates a hypothesis that expresses a relationship between domain-specific concepts and can be tested empirically. Experiments or surveys are conducted by measuring the variables or testing the conditions posited in the hypothesis, and the results are reported together with the empirical methods and the tested hypothesis in a scientific publication. In such scientific settings, a carefully developed and thought-through hypothesis (which we see as Toulmin's [1] "claim" in a scientific context) is at the core of the argumentation process. This hypothesis must be specific enough for a researcher to test it empirically. Still, at the same time, it should also relate to previous general claims made in the community. In actual scientific publications, the relationship between a hypothesis explicitly formulated for the study's context and the general claim it is based on is often neither made explicit nor obvious [2]. Also, hypotheses are usually given as complex statements that include scientific and colloquial terms, and the meaning of both can be ambiguous [3]. For instance, the term "resistance" is used with slightly different meanings by different authors, even within a given domain, and terms like "often" are interpreted differently by different readers. Consequently, scientific hypotheses are a challenging case for modeling, as workflows are required for aligning complex claims with generic structures while at the same time leaving room for the inclusion of domain-specific concepts and knowledge.

While some suggestions for modeling scientific hypotheses already exist (see Sect. 2), they are usually hardly accessible to scientists outside the argumentation community. On the other hand, for experts in formal argumentation, computational linguistics, and semantic modeling, it is not always obvious how best to connect the available tools and approaches to workflows in empirical sciences. A solution to this challenge is the formation of interdisciplinary teams. With this publication, we want to share results from a project that brought together domain experts (in this case, invasion biologists) with experts from semantic modeling and computational linguistics [4]. Our project aims to explore how natural language processing (NLP) and semantic modeling can be leveraged to enhance workflows in scientific research. More specifically, our long-term goal is the automated synthesis of research results testing scientific hypotheses in invasion biology and other domains. To achieve this goal, it is necessary to develop methods for linking scientific papers reporting on empirical tests to major hypotheses relevant to the respective domain.

A prerequisite for such an automated linking of empirical tests to hypotheses is the formalization of hypothesis statements. In this paper, we introduce a framework for transferring hypotheses given in scientific papers in the form of natural language statements into more formalized statements. We use examples from the domain of invasion biology to demonstrate how the framework can help clarify the relationships between the general hypotheses put forward in scientific debates and specific, complex hypotheses directly relating to empirical studies. This paper aims to report on our interdisciplinary efforts to combine domain-specific knowledge of needs and challenges with expert knowledge of tools and approaches from semantic modeling and NLP. The resulting framework is meant as a guideline to be used by experts in a scientific domain who work on synthesizing the knowledge of their field.

In the following, we first give an overview of related work. Next, we introduce our working example and use that to introduce our suggestion for moving towards a formalization of scientific hypotheses. We then report on ongoing applications of the framework. We point out the limitations of our approach and close with an outlook.

2 Related Work

Our suggestions are based on and related to past and ongoing work in the fields of argumentation modeling, knowledge representation, and invasion biology.

2.1 Argumentation Modeling for Complex Scientific Claims

Argumentation is studied in different fields and disciplines, like philosophy, computer science, computational linguistics, and more domain-oriented disciplines like biology. Especially in philosophy, computational linguistics, and NLP, a common approach is to develop abstract representations of arguments and argumentation processes to understand communication processes and how dissent and consensus form. In this context, "toy arguments" are often used to demonstrate the applicability of the respective abstract and formalized argumentation schemes (e.g., [1, 5]). A complementary approach uses AI-based tools for mining arguments in large amounts of data containing informal, primarily textual statements of real-world arguments (see this survey: [6]). While formal accounts are often difficult to apply and to scale up to complex real-world arguments, data-driven argument mining usually does not account for formal aspects of arguments formulated in text.

Regarding formal argumentation analysis, few studies have focused on scientific literature. One example is [7], where the authors suggest an explanatory argumentation framework (EAF) for representing argumentation processes among scientists. In that case, the goal was to model the conceptual structure of the main arguments brought forward by different agents in a scientific debate. The focus of this approach is not so much on the relationship between general and specific claims, nor is the aim to guide hypothesis formulation or identifying hypotheses in texts.

2.2 Knowledge Representation: Modeling Scientific Language with Knowledge Graphs

Semantic Web techniques provide ways to formalize knowledge. On the one hand, this allows machines to act on information; on the other hand, this supports humans in providing concrete representations (e.g., making hidden assumptions and subtle differences in understanding explicit). Knowledge graphs are one such approach that is widely regarded as very promising. They are successfully used in industry but also in scientific settings. In knowledge graphs, nodes represent entities of interest, while edges represent relations between these entities. The graphs are encoded in a (typically machine-actionable) graph data model [8]. One example of their application to model scientific language is [9]. They suggest representing evidence from empirical studies in neuroscience in the form of Research Maps[1]. Here, hypothesized causal relationships are represented as directed graphs, where each node gives the identity and properties of a biological phenomenon. Experimental evidence can be fed into the graphs, allowing to visually represent alignment or disagreement between hypotheses and evidence. This approach, however, focuses on representing the results of empirical work. Consequently, the scheme does not allow for clarifying hierarchical links between complex hypotheses tailored to empirical settings and general, major claims. Also, the aim is to provide templates that researchers can fill out to report their results in a machine-actionable format; the framework is not intended to enhance argument analysis in textual publications.

With a specific focus on formalizing scientific hypotheses, [10] suggested the DISK framework and ontology. DISK was designed to enable automated discovery, hypothesis testing, and revision. As in the case of the Research Maps framework, the focus is on modeling results from empirical studies. Therefore, modeling hierarchical relationships between hypotheses is not straightforward in this setting. Also, as far as we know, the framework has not been implemented and used. It remains unclear how DISK could be used to discover complex versions of hypotheses in actual scientific publications.

Since natural language hypothesis statements can be pretty complex, a stepwise approach towards formalization is practical. The AIDA language suggested by [11] offers a first-step method. This method translates natural language statements into atomic, independent, declarative, and absolute sentences. Such sentences can then derive valid nodes in a knowledge graph.

2.3 Hypothesis Representation in Invasion Biology

Invasion biology studies human-induced transport, introduction, establishment, spread, and impact of organisms. Due to global transport and trade, many species have been translocated to areas outside their natural range [12]. Research in this field is concerned with identifying mechanisms of invasions, often motivated by the goal of developing management solutions. The field is of particular interest in the context of argumentation because numerous major hypotheses have been formulated over time on why species can establish and spread [13–15] (Table A1). This allows for identifying sets of scientific publications that argue for or against one of these hypotheses [16]. Such sets can then be used to develop and test methods for argumentation analysis [17].

In previous work, Heger, Jeschke, and colleagues suggested the hierarchy-of-hypotheses (HoH) approach, according to which scientific hypotheses can be represented as hierarchies [2, 16]. In an HoH, a broad, general claim is given as an overarching hypothesis at the top level, which branches out into more specific versions or sub-hypotheses forming the lower levels. These sub-hypotheses either specify how research on that overarching question has been implemented (‘operational hypotheses’) or represent conceptual refinements, which can be either specification (e.g., spelling out factors that could have caused an effect) or decompositions (e.g., illustrating the partial arguments contained in a broad claim). Concerning the latter, [18] has suggested that it

[1] https://researchmaps.org/.

can be helpful to represent mechanistic hypothesis refinements as causal network diagrams. Decomposition then means adding nodes to a causal chain or network. In the following, we build on these ideas for a stepwise formalization of complex scientific claims.

3 Example: The Biotic Resistance Hypothesis

To demonstrate the challenges connected to treating hypotheses as complex claims in a real-world setting, we give an example of one of the major hypotheses suggested as a potential explanation for the successful establishment and spread of invasive species, namely the Biotic Resistance Hypothesis. In its general version, this hypothesis posits that "*An ecosystem with high biodiversity is more resistant against non-native species than an ecosystem with lower biodiversity*" [19]. In scientific papers, however, such a general formulation is rarely used [17]. Instead, authors of scientific papers tend to use formulations that directly account for the particular case they chose to study and the specific experimental setting. For example, a publication presenting results from empirical tests of the Biotic Resistance Hypothesis used the following formulations: "*…species already in the community with similar functional traits to those of the invaders should have the greatest competitive effect on invaders.*" "*We used experimental communities in a serpentine grassland in California, USA, to assess the extent to which […] functional diversity influenced success of two different types of invading plants: early-season annuals (E) and late-season annuals (L)[…]*". [20].

Such complex statements, differing significantly from the general claim they relate to, can be pretty hard to identify for standard NLP classifiers [17]. Even for scientists, at least those not familiar with the respective claim and underlying theory (e.g., freshly starting Ph.D. students), the link of these complex statements to the major hypothesis is often hard to recognize. An argumentation machine assisting the understanding of such complex claims and aiding the development of own related hypotheses would therefore be helpful [4]. However, this requires developing a framework for formalizing scientific hypotheses and clarifying links between general and specific hypothesis formulations. In the following, we present a suggestion for such a framework.

4 Towards Formalizing Scientific Hypotheses

Our suggestion involves several steps (Fig. 1). Natural language statements of general hypotheses are reformulated into AIDA statements [11] by domain experts. These statements are subsequently translated into further formalized statements of the form subject–relationship–object. A classification scheme allows linking the general statement to the specific claims, and ontologies specify their components. Further, NLP classifiers are used to identify general and specific hypothesis statements in texts (this step is described in [17, 21]).

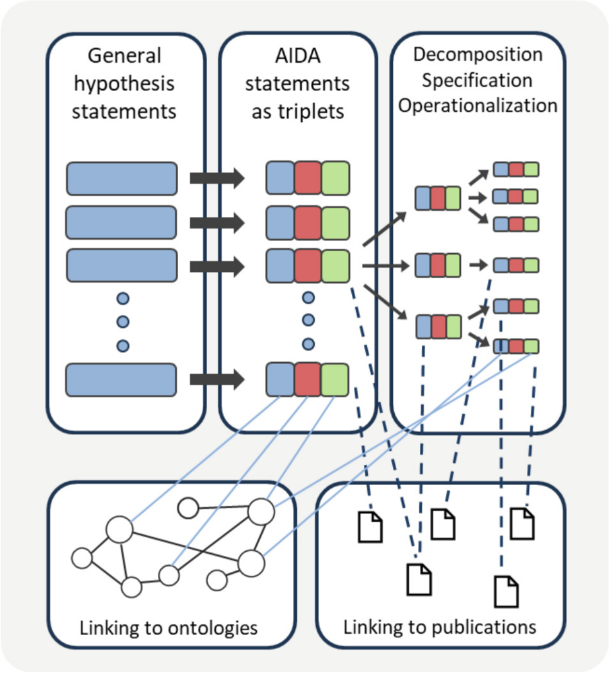

General hypothesis statements
AIDA statements as triplets
Decomposition Specification Operationalization
Linking to ontologies
Linking to publications

Fig. 1. Suggested workflow for developing semi-formal hypothesis statements and clarifying links between general hypotheses and hypothesis statements in scientific texts.

4.1 A Generic Structure for Scientific Hypotheses

Moving towards a formalized representation of scientific hypotheses in invasion biology, starting with broad, major hypotheses, is helpful because these are usually less complex than the refined versions formulated in papers reporting on empirical tests. Taking ten major hypotheses in invasion biology as examples, the invasion biology experts amongst the author group translated the textual versions (Table A1) into AIDA statements, following the methodology suggested in [11]. From these, the domain experts developed formalized versions consisting of a subject, an object, and a hypothesized relationship between these two (analogously to the familiar format of subject-predicate-object triples in edge-labeled graphs, e.g., knowledge graphs encoded in the RDF data model). The subject and the object are often complex in themselves, and we introduced further formalization by distinguishing the core variable, a qualifier for cases in which the core variable has qualitatively distinct states, and a term giving further context concerning settings in which the statement holds (Table A2).

4.2 Linking Hypothesis Formulations to Semantic Models

A critical element in moving from natural language formulations of hypotheses to formalized statements is linking the constituting concepts to entities in machine-actionable ontologies. We suggest using the SuperPattern Ontology[2] [22] to model the hypothesized relationships between subject and object as well as the qualifiers. It contains a set of relations useful for describing causal relationships (e.g., "*contributes to*", "*prevents*", "*inhibits*") and comparisons (e.g., "*has smaller value than*", "*has larger value than*").

Some invasion biology hypotheses are initially given in a comparative form. This is the case for the Biotic Resistance Hypothesis but also for Darwin's Naturalization Hypothesis, the Disturbance Hypothesis, the Island Susceptibility Hypothesis, the Limiting Similarity Hypothesis, and the Phenotypic Plasticity Hypothesis (Table A1). The underlying ideas, however, refer to causal relationships. In these cases, we suggest that both variants can be helpful, the comparative version that is close to the original textual definition and an additional causal version referring to the underlying causal reasoning (Table A2). We think of the comparative versions as some kind of operationalization: In an empirical setting, comparative claims are usually easier to test than causal claims since the former do not necessarily demand to implement experiments. We suggest formalizing the causal variants of the hypotheses in such a way that the subject always gives the invasion driver, i.e., the factor hypothesized to be the underlying force behind a biological invasion or its impacts. The object describes the expected invasion outcome.

As Table A2 demonstrates for the ten hypotheses, the variables and the terms giving context for each subject and object are complex, with little overlap in the used concepts or terms (an exception being "*invasion success*"). This mirrors the complexity of the scientific field of invasion biology, with many potentially influential factors. We, therefore, chose a stepwise approach for modeling them in an ontology created explicitly for this purpose, i.e., the Invasion Biology Ontology INBIO [23]. First, we obtained expert opinion to identify core terms in each of the ten hypotheses. For the Biotic Resistance Hypothesis, these terms were "*ecosystem*", "*biodiversity*" and "*species*". Next, we searched for existing ontologies containing these terms; where this was successful, we used a fusion/merge strategy to integrate respective modules into the INBIO [24]. In further steps, more concepts have been added to provide full conceptual models of the subjects and objects of the ten hypotheses.

The suggested generic structure does not necessarily capture the structure of all scientific hypotheses, but we suggest it can be beneficial for hypotheses describing causal relationships. Hypotheses representing generalized statistical claims (descriptive or statistical hypotheses [25]) do not necessarily follow this form. In our set of ten hypotheses, this was the case for the Tens Rule, which posits that "*Approximately 10% of species successfully take consecutive steps of the invasion process*" (Tables A1 and A2).

4.3 Classifying Relationships Between General and Specific Claims

The previous two subsections have described steps toward formalizing broad, overarching hypotheses. A next step that we consider necessary for linking these formalized versions of major hypotheses to actual hypothesis statements in publications reporting on empirical tests is to clarify the relationship between the overarching hypotheses and the refined sub-hypotheses. Building on the HoH approach, we suggest treating every

[2] https://larahack.github.io/linkflows_superpattern/doc/sp/index-en.html.

hypothesis as a component of a hierarchical system with ‘parents’ and ‘children’. As described in [2], we recommend distinguishing between three kinds of refinements: (A) decomposition, (B) specification, and (C) operationalization (Fig. 2).

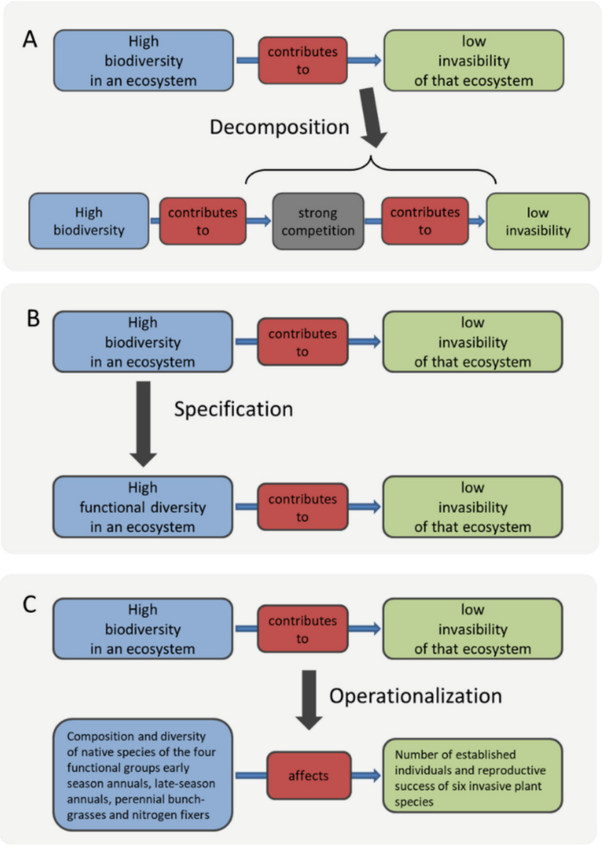

A
High biodiversity in an ecosystem
contributes to
low invasibility of that ecosystem
Decomposition
High biodiversity
contributes to
strong competition
contributes to
low invasibility
B
High biodiversity in an ecosystem
contributes to
low invasibility of that ecosystem
Specification
High functional diversity in an ecosystem
contributes to
low invasibility of that ecosystem
C
High biodiversity in an ecosystem
contributes to
low invasibility of that ecosystem
Operationalization
Composition and diversity of native species of the four functional groups early season annuals, late-season annuals, perennial bunch-grasses and nitrogen fixers
affects
Number of established individuals and reproductive success of six invasive plant species

Fig. 2. Three approaches for relating general versions of scientific hypotheses to more specific ones, demonstrated with the example of the Biotic Resistance Hypothesis in invasion biology: (A) decomposition, (B) specification, and (C) operationalization. See the main text for more information.

With decomposition, we denote the process of making those causal relationships explicit, which are implicit parts of the reasoning behind a hypothesis (see [18] for a worked example of the Enemy Release Hypothesis). Coming back to the Biotic Resistance Hypothesis, the general definition points out the *negative effects of high biodiversity on invasion success*, whereas [20] hypothesizes a *competitive effect of native species on invaders*. An expert in invasion biology can draw from background knowledge to make the connection. For such an expert, it will be evident that intense competition affects invasion success. The refinement of the Biotic Resistance Hypothesis in [20] thus adds nodes to the hypothesized causal graph, making more of the hypothesized mechanism explicit (Fig. 2A).

In the above example, the authors additionally applied the specification strategy. Specifying a hypothesis involves exchanging the nodes of the hypothesized causal chain or network with more concrete versions (Fig. 2B). In the cited example, instead of testing for a general effect of high biodiversity on the chosen invasive species, the authors tested for functional diversity effects. By functional diversity, the authors meant the presence or absence of plant species representing one of four groups that differ in their ecological behavior, namely early-season plants with an annual life cycle, late-season species with an annual life cycle, grasses growing in bunches and living longer than one year, and herbs with the ability to fix atmospheric nitrogen.

The third possibility in which a specific version of a hypothesis can be linked to its general version is operationalization. To operationalize a hypothesis means to describe what exactly will be empirically tested. In the described case, the authors chose to examine the effects of manipulating the composition and diversity of native species of the four functional groups (early-season annuals, late-season annuals, perennial bunchgrasses, and nitrogen fixers). Their dependent variable or 'object' was the number of established individuals and the reproductive success of six selected invasive plant species from those groups (Fig. 2C).

The described operations can also be applied in the other direction. For example, a hypothetical complex causal chain or network can be simplified, which would be the inverse of decomposition (Fig. 2A). An existing hypothesis, perhaps derived from studying a specific context, can be generalized to a broader context (e.g., in terms of taxa or life stages covered, geographic range or other ecological gradients); this would be the opposite of specification (Fig. 2B). Finally, from a hypothesis generated, e.g., from an empirical observation under specified experimental conditions, a broader, more abstract version can be derived; such an abstraction would be the opposite of an operationalization (Fig. 2C).

The suggested scheme can be a basis for linking actual hypothesis statements in publications reporting on empirical tests to major, more general hypotheses [2, 16]. For example, in their literature review on the Biotic Resistance Hypothesis, [19] identified 15 empirical studies that focused on functional diversity as a specific form of biodiversity, whereas 126 empirical tests in their dataset instead studied species richness, which is a different specification of biodiversity.

To allow for the implementation of the framework in the context of argumentation analysis, we are currently developing a Hypothesis Ontology containing the concepts identified as hypothesis components and the possible relationships between general and specific variants, as just described. Figure 3 depicts the already developed modeling of types of entities and their relationships; adding concrete instances belonging to these types (e.g., the Biotic Resistance Hypothesis as one specific Hypothesis) is ongoing work. In this model, a Hypothesis is linked to a HypothesisDefinition. The definition "*An ecosystem with high biodiversity is more resistant against non-native species than an ecosystem with lower biodiversity*" [19] will be one instance of the type HypothesisDefinition. The distinction between the Hypothesis and the HypothesisDefinition is necessary, as several subtly different definitions exist for many high-level hypotheses. Each of these definitions is further captured in a HypothesisStatement. We model HypothesisStatements as SuperPatternInstances [22]. They possess a Label, Context, Subject, Relations, Objects, and Qualifiers. Subjects and Objects can be complex and consist of Qualifiers, Variables, and Contexts. Hypothesis and HypothesisDefinitions can have subclass relationships to reflect the hierarchical structure described above. A Hypothesis can be supported (or refuted) by Evidence and equipped with Provenance as defined in the Prov-O ontology[3].

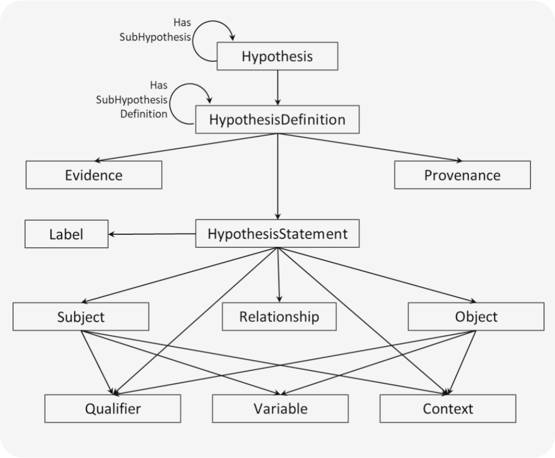


Fig. 3. Conceptual scheme for the Hypothesis Ontology

5 Applications of the Framework

The current situation in which major scientific research results are mainly published in PDF format hinders the integration of AI technology in scientific workflows [26]. An important step towards overcoming this barrier would be to enrich the bibliographic meta-data of scientific publications with machine-readable information about the publications' content, including studied hypotheses. The suggested framework and related semantic modeling can provide a basis for such endeavors. We are currently exploring two parallel pathways in this direction. The first of these pathways involves using Wikidata to link entries about publications to entries about hypotheses, while the second introduces hypotheses as a publication type in its own rights.

The Wikidata pathway builds on community curation workflows under the umbrella of the WikiCite initiative that collects bibliographic metadata in Wikidata [27]. It further involves the development of tools for exploring the resulting knowledge graph (e.g. [28]). In this context, we regularly identify invasion biology publications and annotate them as such, with additional workflows to annotate the identified publications for author

[3] https://www.w3.org/TR/prov-o/.

disambiguation, main subjects, or methods used. For each hypothesis to be used in this workflow, a dedicated Wikidata entry is required, and we have created such entries for the most common hypotheses in invasion biology, including those listed in Table A1. These entries can then be annotated, e.g., in terms of the publications from which they originated or the concepts they relate to. The aim is to establish links between the hypotheses and scientific publications testing or discussing them. In the future, this will allow for better findability of relevant publications in an Open Science environment and options for on-demand meta-analyses [29].

For the second pathway, we developed a scheme for a new publication type - Hypothesis Descriptions [30]. Such Hypothesis Descriptions are aimed at formalizing how invasion biology hypotheses are described (especially in terms of which concepts and relationships they cover) and how differences between hypothesis variants can be expressed, both for humans and in a machine-actionable fashion. This scheme is pioneered in the open-science journal Research Ideas and Outcomes [31] and builds on the nanopublication standards beginning to be adopted in biodiversity-related publications [32].

In the context of invasion biology (and other fields of science), the suggested framework can further be used as a guideline for formulating hypotheses. In invasion biology, the ambiguity of hypothesis formulations is often considered challenging (see, e.g., [33]). Still, it is not an established practice to carefully consider the relationship between a specific, complex claim made in a publication and the general version it has been derived from or to use consistent language for formulating hypotheses. We suggest that our framework could offer guidance, thus enhancing research efficiency. For example, in the case of the Enemy Release Hypothesis, empirical research so far has mainly focused on only one of its components [33, 34]. However, to establish whether or not this hypothesis can be regarded as a reasonable explanation for invasion success, it would be necessary to study the complete hypothesized causal chain. Such gaps are more easily identified if respective publications clarify which kind of hypothesis refinement is chosen for the study context.

6 Limitations

Invasion biology, the research domain we used to develop our framework, is a relatively straightforward example because the domain is characterized by many explicitly formulated major hypotheses repeatedly synthesized by the scientific community [13–16]. In other disciplines, it might be much harder to even identify such general claims. For the neighboring discipline of urban ecology, [35] demonstrated how similar lists of major hypotheses can be collated with a combination of expert involvement and literature analyses. This general approach can, in principle, be applied to any other scientific domain. Also, we believe that the NLP models we develop based on the introduced hypothesis formalization can be used later for automatic/semi-automatic hypothesis discovery in other fields as well. The suggestion for linking specific formulations of empirical tests to general claims is also not limited to an application in invasion biology. [36] demonstrated how specific claims in medicine can be linked to a general, major claim by specification and operationalization. Still, future work is needed to clarify for which scientific domains it is possible and useful to implement all steps towards hypothesis formalization outlined above.

Currently, it is an open question how our ontology-based, multi-level formalization of hypotheses can feed into NLP-based argument mining methods, i.e., hypothesis identification in particular [21]. While much recent work is on integrating language modeling and knowledge graphs, it is unclear how these methods scale to the complex problem of hypothesis identification in scientific papers, which requires deep semantic reasoning and domain-specific knowledge. In future work, relevant ontologies will be integrated with text-driven approaches to argument mining and enhance the implicit knowledge in language modeling-based approaches with explicit knowledge. This can be achieved, for instance, with recent methods for so-called "knowledge injection into language models", see [25].

Moving towards formalizing scientific hypotheses requires exchanging complex natural language with streamlined and unified terms and concepts. It is necessary to carefully study under which conditions the gain of formalizing outweighs the potential information losses during this process. This challenge can become even more demanding once the semi-formal statements suggested in Table A2 are further transformed, e.g., into logical statements that provide a foundation for automated reasoning. An annotation study could be a practical next step to help clarify how well our proposed scheme can capture complex hypothesis statements in actual scientific texts.

7 Conclusions and Outlook

In this article, we suggested a framework for moving towards a formalization of scientific hypotheses and clarifying links between general and specific hypothesis formulations. Developing the framework was an interdisciplinary effort, considering knowledge from invasion biology, philosophy of science, computational linguistics, and semantic modeling. We suggest our framework can be helpful for argumentation analysis in scientific publications. Further, it can help in taking steps towards reprocessing scientific publications and making published research available for AI-based analyses. Finally, the framework can guide researchers during the hypothesis formulation process. We suggest that domain experts can directly profit from our framework because it motivates to make intuitions explicit and fosters conceptual analysis, which can directly benefit the quality of scientific work [37].

Therefore, implementing the framework as a user interaction tool is an essential next step. A prototype of such a tool already exists, and a first version will soon be available at hi-knowledge.org[4]. The tool will help researchers identify major invasion hypotheses in texts, link to background information necessary for understanding technical terms, and, in the future, offer guidance to formulate their own specific and complex research hypothesis tailored to the focal empirical setting. Implementing AI-based tools in all steps of the scientific workflow is a timely and urgent need. This would significantly enhance efficiency [38] and allow for better utilization of knowledge gained in research for solving current societal challenges. We hope our framework will motivate and facilitate innovative steps in this direction.

[4] https://hi-knowledge.org/.

Acknowledgements. This work was funded by Deutsche Forschungsgemeinschaft DFG (project number 455913229; T.H., M.B., J.M.J., B.K-R, S.Z.) and the VolkswagenStiftung (97 863; T.H., J.M.J., D.M.). We thank Maud Bernard-Verdier, Camille Musseau, and Florencia Yannelli for their comments on the formalization framework. Three anonymous reviewers and a meta-reviewer provided comments that helped us to improve the text.

Appendix

Table A1. Ten major hypotheses in invasion biology and their textual definitions as given in [39].

Hypothesis	Acronym	Definition
Biotic resistance hypothesis	BR	An ecosystem with high biodiversity is more resistant against non-native species than an ecosystem with lower biodiversity
Darwin's naturalization hypothesis	DN	Invasion success of non-native species is higher in areas that are poor in closely related species than in areas that are rich in closely related species
Disturbance Hypothesis	DS	Success of non-native species is higher in highly disturbed than in relatively undisturbed ecosystems
Enemy release Hypothesis	ER	The absence of enemies in the exotic range is a cause of invasion success
Invasional meltdown hypothesis	IM	The presence of non-native species in an ecosystem facilitates invasion by additional species, increasing their likelihood of survival or ecological impact
Island susceptibility hypothesis	IS	Non-native species are more likely to become established and have major ecological impacts on islands than on continents
Limiting similarity hypothesis	LS	Success of non-native species is high if they strongly differ from native species, and it is low if they are similar to native species
Phenotypic plasticity Hypothesis	PH	Invasive species are more phenotypically plastic than non-invasive or native ones
Propagule pressure hypothesis	PP	High propagule pressure (a composite measure consisting of the number of individuals introduced per introduction event and the frequency of introduction events) is a cause of invasion success
Tens rule	TEN	Approximately 10% of species successfully take consecutive steps of the invasion process

Table A2. Semi-formalized representations of ten major hypotheses in invasion biology. For hypotheses stated as comparisons (Table A1; relationship "has larger value than"), a causal variant is also given. In the causal hypothesis variants, the subject describes the hypothesized driver and the object of the invasion outcome. H: Hypothesis, Q: Qualifier. For acronyms, see Table A1.

H	Subject			Relationship	Object		
	Q	Variable	Context		Q	Variable	Context
BR		Biodiversity	in an ecosystem resistant against non-native species	has larger value than		biodiversity	in an ecosystem with low resistance
	High	biodiversity	in an ecosystem	contributes to	low	invasibility	of that ecosystem
DN		Invasion success	in ecosystems poor in closely related species	has larger value than		invasion success	in ecosystems rich in closely related species
	Low	number of species closely related to a non-native species	in an ecosystem	contributes to	high	invasion success	of this species in this ecosystem
DS		Invasion success	in highly disturbed ecosystems	has larger value than		invasion success	in relatively undisturbed ecosystems
	High	disturbance	of an ecosystem	contributes to	high	invasion success	of non-native species in that ecosystem
ER	No	enemies	of a species in its non-native range	contributes to	high	invasion success	of this species in the new range
IM		Invasion success	of previously arriving non-native species	enables		invasion success or impact	of new non-native species
IS		Invasion success and impact of non-native species	on islands	has larger value than		Invasion success and impact of non-native species	

(*continued*)

Table A2. (*continued*)

H		Subject		Relationship		Object	
	Q	Variable	Context		Q	Variable	Context
		Arrival on island and not continental land		contributes to	high	invasion success and impact	
LS		Invasion success	in ecosystems poor in functionally similar species	has larger value than		invasion success	in ecosystems rich in functionally similar species
	High	functional similarity to native species	of invasive species in an ecosystem	contributes to	low	invasion success	of that species in that ecosystem
PH		Phenotypic plasticity	of invasive species	has larger value than		phenotypic plasticity	of non-invasive or native species
	High	phenotypic plasticity	of a non-native species	contributes to	high	invasion success	of this species
PP	High	propagule pressure	of a species in its non-native range	contributes to	high	invasion success	of this species in that area
TEN		n/a	n/a	n/a		n/a	n/a

References

1. Toulmin, S.E.: The Uses of Argument, 2 edn. Cambridge University Press, Cambridge (2003). https://doi.org/10.1017/CBO9780511840005
2. Heger, T., et al.: The hierarchy-of-hypotheses approach: a synthesis method for enhancing theory development in ecology and evolution. Bioscience **71**(4), 337–349 (2021). https://doi.org/10.1093/biosci/biaa130
3. Madin, J.S., Bowers, S., Schildhauer, M.P., Jones, M.B.: Advancing ecological research with ontologies. Trends Ecol. Evol. **23**(3), 159–168 (2008). https://doi.org/10.1016/j.tree.2007.11.007
4. Heger, T., Zarrieß, S., Algergawy, A., Jeschke, J.M., König-Ries, B.: INAS: interactive argumentation support for the scientific domain of invasion biology. Res. Ideas Outcomes **8**, e80457 (2022). https://doi.org/10.3897/rio.8.e80457
5. Walton, D., Reed, C., Macagno, F.: Argumentation Schemes. Cambridge University Press, Cambridge (2008). https://doi.org/10.1017/CBO9780511802034
6. Lawrence, J., Reed, C.: Argument mining: a survey. Comput. Linguist. **45**(4), 765–818 (2020). https://doi.org/10.1162/coli_a_00364
7. Šešelja, D., Straßer, C.: Abstract argumentation and explanation applied to scientific debates. Synthese **190**(12), 2195–2217 (2013). https://doi.org/10.1007/s11229-011-9964-y
8. Hogan, A., et al.: Knowledge graphs. ACM Comput. Surv. **54**(4), 1–37 (2021). https://doi.org/10.1145/3447772

9. Matiasz, N.J., et al.: ResearchMaps.org for integrating and planning research. PLoS ONE **13**(5), e0195271 (2018). https://doi.org/10.1371/journal.pone.0195271
10. Garijo, D., Gil, Y., Ratnakar, V.: The DISK hypothesis ontology: capturing hypothesis evolution for automated discovery (2017)
11. Kuhn, T.: Using the AIDA language to formally organize scientific claims. In: Wyner, A., Davis, B., Keet, C.M. (eds.) Controlled Natural Language: Proceedings of the 6th International Workshop, CNL. Frontiers in Artificial Intelligence and Applications, pp. 52–60 (2018). https://doi.org/10.3233/978-1-61499-904-1-52
12. Roy, H.E., et al.: IPBES Invasive Alien Species Assessment: Summary for Policymakers (Version 2). Zenodo (2023).https://doi.org/10.5281/zenodo.8314303
13. Enders, M., et al.: A conceptual map of invasion biology: Integrating hypotheses into a consensus network. Glob. Ecol. Biogeogr. **29**, 978–991 (2020). https://doi.org/10.1111/geb.13082
14. Catford, J.A., Jansson, R., Nilsson, C.: Reducing redundancy in invasion ecology by integrating hypotheses into a single theoretical framework. Divers. Distrib. **15**(1), 22–40 (2009). https://doi.org/10.1111/j.1472-4642.2008.00521.x
15. Daly, E.Z., et al.: A synthesis of biological invasion hypotheses associated with the introduction–naturalisation–invasion continuum. Oikos **2023**(5), e09645 (2023). https://doi.org/10.1111/oik.09645
16. Jeschke, J.M., Heger, T. (eds.): Invasion Biology: Hypotheses and Evidence. CAB International, Wallingford, UK (2018)
17. Brinner, M., Heger, T., Zarriess, S.: Linking a hypothesis network from the domain of invasion biology to a corpus of scientific abstracts: the INAS dataset. In: Proceedings of the First Workshop on Information Extraction from Scientific Publications, pp. 32–42. Association for Computational Linguistics (2022)
18. Heger, T.: What are ecological mechanisms? Suggestions for a fine-grained description of causal mechanisms in invasion ecology. Biol. Philos. **37**(2), 9 (2022). https://doi.org/10.1007/s10539-022-09838-1
19. Jeschke, J.M., Debille, S., Lortie, C.J.: Biotic resistance and island susceptibility hypotheses. In: Jeschke, J.M., Heger, T. (eds.) Invasion Biology Hypotheses and Evidence, pp. 60–70. CAB International, Wallingford, UK (2018)
20. Hooper, D.U., Dukes, J.S.: Functional composition controls invasion success in a California serpentine grassland. J. Ecol. **98**(4), 764–777 (2010). https://doi.org/10.1111/j.1365-2745.2010.01673.x
21. Brinner, M., Zarrieß, S., Heger, T.: Weakly supervised claim localization in scientific abstracts. In: RATIO-24, Bielefeld, Germany, pp. 20–38. Springer, Heidelberg (2024)
22. Bucur, C.-I., Kuhn, T., Ceolin, D., van Ossenbruggen, J.: Expressing high-level scientific claims with formal semantics. In: Proceedings of the 11th International Conference on Knowledge Capture Conference, K-CAP 2021, New York, NY, USA, pp. 233–40. Association for Computing Machinery (2021). https://doi.org/10.1145/3460210.3493561
23. Algergawy, A., Gänßinger, M., Heger, T., Jeschke, J., König-Ries, B.: The Invasion Biology Ontology (INBIO) [Data set]. Zenodo (2022).https://doi.org/10.5281/zenodo.6826848
24. Algergawy, A., Stangneth, R., Heger, T., Jeschke, J.M., König-Ries, B.: Towards a core ontology for hierarchies of hypotheses in invasion biology. In: Harth, A., et al. (eds.) ESWC 2020. LNCS, vol. 12124, pp. 3–8. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-62327-2_1
25. Betts, M.G., et al.: When are hypotheses useful in ecology and evolution? Ecol. Evol. **11**(11), 5762–5776 (2021). https://doi.org/10.1002/ece3.7365
26. Kuhn, T., Dumontier, M.: Genuine semantic publishing. Data Sci. **1**, 139–154 (2017). https://doi.org/10.3233/DS-170010

27. Wyatt, L., et al.: WikiCite 2020–2021: citations for the sum of all human knowledge. Zenodo (2021).https://doi.org/10.5281/zenodo.5363757
28. Nielsen, F.Å., Mietchen, D., Willighagen, E.: Scholia, scientometrics and Wikidata. In: Blomqvist, E., et al. (eds.) ESWC 2017. LNCS, vol. 10577, pp. 237–259. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70407-4_36
29. Jeschke, J.M., Heger, T., Kraker, P., Schramm, M., Kittel, C., Mietchen, D.: Towards an open, zoomable atlas for invasion science and beyond. NeoBiota **68**, 5–18 (2021). https://doi.org/10.3897/neobiota.68.66685
30. Heger, T., Jeschke, J.M., Bernard-Verdier, M., Musseau, C.L., Mietchen, D.: Hypothesis description: enemy release hypothesis. Res. Ideas Outcomes **10**, e107393 (2024). https://doi.org/10.3897/rio.10.e107393
31. Mietchen, D., Mounce, R., Penev, L.: Publishing the research process. Res. Ideas Outcomes **1**, e7547 (2015). https://doi.org/10.3897/rio.1.e7547
32. Penev, L., et al.: Nanopublications for biodiversity go live. Biodivers. Inf. Sci. Stan. **7**, e110725 (2023). https://doi.org/10.3897/biss.7.110725
33. Brian, J., Catford, J.: A mechanistic framework of enemy release. Ecol. Lett. **26**(12), 2147–2166 (2023). https://doi.org/10.1111/ele.14329
34. Heger, T., Jeschke, J.M.: Enemy release hypothesis. In: Jeschke, J.M., Heger, T. (eds.) Invasion Biology Hypotheses and Evidence, pp. 92–102. CAB International, Wallingford, UK (2018) https://doi.org/10.1079/9781780647647.0092
35. Lokatis, S., et al.: Hypotheses in urban ecology: building a common knowledge base. Biol. Rev. **98**, 1530–1547 (2023). https://doi.org/10.1111/brv.12964
36. Bartram, I., Jeschke, J.M.: Do cancer stem cells exist? A pilot study combining a systematic review with the hierarchy-of-hypotheses approach. PLoS ONE **14**(12), e0225898 (2019). https://doi.org/10.1371/journal.pone.0225898
37. Guest, O., Martin, A.E.: How computational modeling can force theory building in psychological science. Perspect. Psychol. Sci. **16**(4), 789–802 (2021). https://doi.org/10.1177/1745691620970585
38. Wang, H., Fu, T., Du, Y., Gao, W., Huang, K., Liu, Z., et al.: Scientific discovery in the age of artificial intelligence. Nature **620**(7972), 47–60 (2023). https://doi.org/10.1038/s41586-023-06221-2
39. Jeschke, J.M., Enders, M., Bagni, M., Jeschke, P., Zimmermann, M., Heger, T.: Hi-Knowledge.org, version 2.0 (2020). Available from: https://hi-knowledge.org/

Weakly Supervised Claim Localization in Scientific Abstracts

Marc Brinner[1](✉), Sina Zarrieß[1], and Tina Heger[2]

[1] Computational Linguistics, Department of Linguistics, Bielefeld University, Bielefeld, Germany
{marc.brinner,sina.zarriess}@uni-bielefeld.de

[2] Leibniz Institute of Freshwater Ecology and Inland Fisheries (IGB), Berlin, Germany
t.heger@tum.de

Abstract. We explore the possibility of leveraging model explainability methods for weakly supervised claim localization in scientific abstracts. The resulting approaches require only abstract-level supervision, i.e., information about the general presence of a claim in a given abstract, to extract spans of text that indicate this specific claim. We evaluate our methods on the SciFact claim verification dataset, as well as on a newly created dataset that contains expert-annotated evidence for scientific hypotheses in paper abstracts from the field of invasion biology. Our results suggest that significant performance in the claim localization task can be achieved without any explicit supervision, which increases the transferability to new domains with limited data availability. In the course of our experiments, we additionally find that injecting information from human evidence annotations into the training of a neural network classifier can lead to a significant increase in classification performance.

Keywords: Explainability · Evidence localization · Claim verification

1 Introduction

A claim lies at the center of most scientific publications, as it constitutes the core proposition that is put forth for consideration and is targeted by the presented evidence [19]. Detailed knowledge about these claims addressed in scientific publications is essential for tasks like literature search and scientific claim verification [40], leading to a variety of research targeted at the annotation, recognition and localization of claims in scientific abstracts and full texts (see Sect. 2.1). Despite significant progress being made, the reliance on direct supervision (e.g., [23,41]) often limits the potential of these approaches, since large and high-quality datasets are uncommon in general and not present at all in many specific domains, and since existing models struggle to generalize to different domains [36]. Especially for the task of localizing evidence for claims within a text, the annotation process for creating the dataset is very time-intensive

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 20–38, 2024.
https://doi.org/10.1007/978-3-031-63536-6_2

and thus more costly, which naturally raises the question of whether a weaker supervision signal, that could be quicker, easier and more consistent to annotate, could be sufficient for solving this complex task.

In this study, we explore the possibility of using weak supervision for the task of claim localization in scientific abstracts. The supervision signal is the information about the general presence of a specific claim in a given abstract (i.e., a textual formulation of that claim or a discrete claim label). This information is used to train a standard neural network classifier that is able to verify the presence of such a semantically distinct claim in a given abstract. We then use model explainability approaches to create a rationale for the classification, which therefore selects spans or sentences from the input that constitute the evidence for the given claim. This is, to our knowledge, the first study that explores the sole use of weak supervision for solving this task.

To test our methods, we evaluate them on two datasets of scientific abstracts with annotated evidence. The first one is the INAS dataset [3], a dataset consisting of scientific abstracts from the field of invasion biology, annotated with information about which hypothesis from the field is addressed. Since no evidence annotations are provided by [3], we perform our own annotation study and annotate 750 abstracts with span-level hypothesis evidence. The second dataset is the SciFact dataset [35], which consists of hand-written claims for a set of scientific abstracts, in combination with sentence-level evidence annotations.

To explore the limits of using explainability approaches for evidence localization, we perform an experiment on injecting the information from evidence annotations into the training process of neural network classifiers. A similar approach has been explored by [38], but we are not aware of such techniques being used for claim verification. In our testing, we find that our method is able to drastically increase the classification performance of the resulting classifier.

The rest of our work is structured as follows: In Sect. 2 we provide background knowledge about scientific claim detection as well as the concept of using input optimization for model interpretability, while Sect. 3 will describe the datasets used in this study. Section 4 then explains our approach for localizing claim evidence as well as a method for injecting evidence information into a standard neural network classifier, while Sects. 5 and 6 will detail the corresponding experiments and results. Section 7 concludes this work with final thoughts and remarks.

The code for the experiments conducted in this study is available at github.com/inas-argumentation/claim_localization.

2 Background

2.1 Scientific Claim Detection

Scientific claim detection has its root in the field of general argument mining, which was formally introduced by [22] and is concerned with locating, classifying and linking argumentative components (so-called argumentative discourse units) in a given argumentative text. Based on the general theory of argumentation

[8,22,34] determined the *claim* to be the center of an argument, as it is the core proposition that is put forth for consideration. A claim, by its nature, is not inherently true and requires further substantiation, which is provided by *premises*, i.e., statements that are generally accepted to be true and do not require further support [29].

As scientific texts are argumentative in nature, the field of argument mining naturally extends to the scientific domain. Recognizing the argumentative structure in a scientific text, as well as the main claim in particular, is essential in tasks like literature search and scientific claim verification [40], leading to the creation of a variety of annotation schemes and datasets [1,10,31,32], many of which focus specifically on the scientific claim: [2] creates a detailed annotation scheme that captures the variety of ways a claim can be formulated in a scientific abstract, [35] create the scientific claim verification task by creating a dataset of hand-written scientific claims and by annotating which sentences in a corpus of scientific abstracts supports or refutes them, and [3] focus on a precise semantic categorization of scientific claims by annotating and classifying claims according to a domain-specific hypothesis network.

Given a specific claim, our study addresses the precise localization of evidence for this exact claim in a given scientific abstract. While many approaches have been proposed to solve similar tasks [2,13,23,41], these methods leverage data annotated on sentence level for supervised learning, which can limit their potential due to the rather small available datasets and the unavailability of any annotated data in many domains. Reasons for this lack of data include the need for expert annotators caused by the focus on the scientific domain, the time-intensive annotation process, as well as the complexity of the annotation task even for domain experts [11].

To our knowledge, no method exists that can reliably detect and locate claims in scientific texts without access to a dataset of samples with explicit sentence-level claim annotations, which can be a problem if a model shall be adapted to a new domain without an existing dataset, as performance has been shown to significantly decrease on out-of-domain samples [36]. Our study aims at closing this gap by creating an approach that only requires weak supervision in the form of abstract-level labels, thus drastically reducing the time and cost needed to create a training set for a new domain.

2.2 Input Optimization for Model Interpretability

For many datasets, evidence annotations for specific claims constitute a rationale for a corresponding classification (e.g., for the claim verification task [35], claim evidence is an explanation for an abstract-level validity label). This characterization of claim evidence creates a natural connection to the field of model interpretability, which is concerned with creating explanations for decisions (e.g., classifications) of black-box machine learning models like neural networks. In the field of natural language processing, explanations for classifications often take the form of individual scores assigned to each input token, with a higher score indicating an increased significance of that token for the predicted score. While a variety of methods have been proposed [20], we will focus on a recent study by [4], as their method called MaRC (Mask-based Rationale Creation) is specifically designed to extract longer, consecutive spans of text as explanations, thus making the explanations better aligned with human reasoning and annotations.

The MaRC approach relies on the concept of input optimization: As neural networks are differentiable, it is possible to calculate the gradient of an objective function with respect to the input features. The MaRC approach uses this concept to remove words from the input by gradually replacing them by *PAD* tokens (in the case of BERT) in a way that maximizes the likelihood of the class that is to be explained, meaning that the words that remain are highly indicative of the respective class.

The MaRC approach assigns parameters w_i and σ_i to each input word x_i, to calculate a mask λ in the following way:

$$w_{i \rightarrow j} = w_i \cdot \exp\big(-\frac{d(i,j)^2}{\sigma_i}\big) \tag{1}$$

$$\lambda_j = \text{sigmoid}(\sum_i w_{i \rightarrow j}) \tag{2}$$

The weight w_i of a word x_i is mainly responsible for its mask value λ_i, but each weight w_i also influences the mask values of the words around it: $d(i,j)$ denotes the distance between two words while $w_{i \rightarrow j}$ denotes the influence of weight w_i towards λ_j. This mask value λ_j is simply calculated by applying the sigmoid to the sum of all influences onto this mask value. This parameterization of the mask, together with an objective function that encourages large values of σ, leads to smooth masks with long consecutive spans of texts being selected. Using this mask, two altered inputs are created:

$$\tilde{x} = \lambda \cdot x + (1 - \lambda) \cdot b \tag{3}$$

$$\tilde{x}^{\mathsf{c}} = (1 - \lambda) \cdot x + \lambda \cdot b \tag{4}$$

b is here an uninformative background (e.g., *PAD* tokens for BERT), meaning that $\tilde{x}$ is created by applying the mask to input x which removes low-scoring words from the input, while $\tilde{x}^{\mathsf{c}}$ applies the reverse mask. The actual objective function that is optimized is the following:

$$\underset{w,\sigma \in \mathbb{R}^n}{\arg\min} \quad -\mathcal{L}(\tilde{x}, c) + \mathcal{L}(\tilde{x}^{\mathsf{c}}, c) + \Omega_\lambda + \Omega_\sigma \tag{5}$$

where we optimize our mask to maximize our class probability of desired class c (given by $\mathcal{L}(\tilde{x}, c)$), meaning that we select words that indicate this class, while minimizing this likelihood for the reverse mask, meaning that words indicative of c will not be masked. The additional regularizers enforce sparsity (Ω_λ) and smoothness (Ω_σ) of the mask. For a more detailed description and derivation, see [4].

3 Datasets

3.1 The INAS Dataset

We evaluate our claim localization approach on the INAS dataset [3]. The dataset consists of 954 paper titles and abstracts from the field of invasion biology, a field concerned with the study of human-induced spread of species outside of their native ranges, caused by factors like global transport and trade. The samples are annotated with labels indicating which of the ten main hypotheses in the field are addressed in a given paper, in combination with an even more fine-grained categorization about specific sub-hypotheses addressed in them, based on a hypothesis network created by [14]. We perform our own annotation study and asked three experts in the field of invasion biology to annotate 750 samples with span-level evidence. The task was to annotate all spans that, to the trained eye, indicate which hypothesis is addressed in the given paper, even if the hypothesis is not explicitly named or stated.

50 samples were annotated by all annotators and we achieved a rather low F1 score of 0.389, indicating that this is a generally challenging annotation task even for domain experts. This is in part caused by one annotator having much lower agreement with the other two, indicating that annotation guidelines were interpreted slightly differently, which, for such a complex task, can quickly reduce agreement scores. The higher F1 score of 0.579 between the other two annotators shows that the general task is well-defined and thus suitable to be tackled by neural networks.

3.2 The SciFact Dataset

We also evaluate our approach on the SciFact dataset [35]. It consists of 5,183 abstracts from a collection of well-regarded journals, in combination with a set of 1,409 hand-written claims that are supported or rejected by papers from the corpus. The papers that verify or reject a claim are annotated on sentence-level with evidence for the respective classification, so that, in contrast to the span-level annotations for the INAS dataset, each sentence completely belongs to the evidence or not.

4 Method

4.1 Span-Level Claim Evidence Localization

We propose a method to perform weakly supervised span-level claim evidence localization. In this setting, we assume the availability of a training set of texts labeled with information indicating which claim (from a fixed set of known claims) is addressed in each of them. Given a text consisting of words $x_1, ..., x_n$, the task of weakly supervised claim localization is now to predict a set $I \subset \{1, ..., n\}$ of indices of words that are part of the ground truth claim evidence annotated by a human annotator. We propose to utilize the MaRC approach (see Sect. 2.2) to solve this task by first training a classifier to perform the claim identification task using only abstract-level labels, which is a standard text classification problem. Afterward, MaRC can be used to create an explanation for the classification of a given sample to produce importance scores for each word in the abstract.

For improved rationale predictions, we propose to perform the optimization from Eq. 5 with respect to several models, but for a single set of mask values. Input optimization is known to overly adapt to the particularities of a given model, which we hypothesize to be mitigated by optimizing with respect to multiple models at once.

4.2 Sentence-Level Claim Evidence Localization

We also propose an approach for sentence-level claim evidence localization. The precise task we consider slightly differs from the one described in the previous section, as here we assume claims to be present in textual form, and to not originate from a fixed set of known claims. Given a claim and an abstract, the task is to predict one of the three labels {*Supports*, *Refutes*, *Not Enough Info*}.

We again start by training a standard text classification model, which now receives the claim and abstract as inputs and predicts one of the three given labels as output. While it would be possible to employ the same procedure as described in Sect. 4.1 and compute sentence scores from the scores for the individual words, this could lead to uncertainties in the case of only very few words in a sentence being selected, as these could be highly important (thus making the whole sentence important) or simple artifacts caused by important words from a neighboring sentence exerting influence.

For this reason, we directly optimize mask weights $w_1, ..., w_n$, with one value being assigned to each input sentence $s_i \in \{s_1, ..., s_n\}$, and define $\lambda_i = \sigma(w_i)$ as the mask value for the sentence. We also alter the interpretation of the mask values λ: Before, each input embedding was linearly blended towards an uninformative embedding, as the input embedding $\tilde{x}_i$ of token i was defined to be $\tilde{x}_i = \lambda_i \cdot x_i + (1 - \lambda_i) \cdot b_i$. Despite good performance of this approach [4], these shifted embeddings constitute out-of-domain inputs as they are not encountered during training, therefore potentially leading to unpredictable behavior of the network. Therefore, we explore the possibility of treating λ as a set of probability distributions, with each λ_i being the parameter of a Bernoulli distribution indicating the probability of sentence s_i belonging to the input. This allows sampling of inputs from this distribution, with each sentence being either completely present or completely removed (replaced by *[PAD]* tokens) in a given sample. We then optimize this distribution to increase the likelihood of samples with high scores according to our objective, leading to the following optimization problem:

$$\underset{w \in \mathbb{R}^n}{\arg\min} \ \ \mathbb{E}_{m \sim \lambda} \left[-\mathcal{L}(\tilde{x}, c) + \mathcal{L}(\tilde{x}^{\mathsf{c}}, c) \right] + \Omega_\lambda \tag{6}$$

where $\tilde{x}$ and $\tilde{x}^{\mathsf{c}}$ are computed using the mask m sampled from λ similarly to Eq. 3 and Eq. 4, but on sentence-level. This equation can not be optimized using standard gradient-descent, as it contains an expectation over a probability distribution. We therefore use the score function estimator [9]:

$$\frac{\partial}{\partial \lambda} \mathbb{E}_{m \sim p(\cdot;\lambda)} \left[f(m) \right] = \mathbb{E}_{m \sim p(\cdot;\lambda)} \left[f(m) \frac{\partial}{\partial \lambda} \log p(m;\lambda) \right] \tag{7}$$

The expectation on the right side can now be approximated by sampling a batch of masks from λ, with $f(m)$ being our likelihood scores for mask m as defined in Eq. 6.

For our specific task, only the sentences from the abstract are masked, while the claim does not receive a mask value to be optimized. Again, we perform the optimization with respect to multiple trained classifiers as further regularization.

4.3 Evidence Injection

While our general methods aim at using weak supervision only, we also explore how far the results can be improved by using evidence annotations in the course of the base classifier training. To do this, we develop a method to inject evidence annotation information into the standard classifier training process. To our knowledge, something remotely similar has only been explored for the case of Support Vector Machines [38]. We test this method on the SciFact dataset and therefore assume the presence of sentence-level evidence annotations.

The altered training paradigm works as follows: Given a training sample x, this sample will be fed three times into the network (all in the same batch). Once in its normal form, once with all evidence sentences removed, and once with all evidence sentences present, but with some other sentences removed. We then train the model to predict the correct label (*Supports* or *Refutes*) for the first and third versions of the sample, but train it to predict the *Not Enough Info* label for the second version. In this way, the classifier learns to differentiate sentences that actually support the claim from sentences that only address the same topic.

5 Experiments

5.1 Span-Level Claim Localization

Experimental Setup. We perform experiments on weakly-supervised span-level evidence localization on the INAS dataset. Given a sample x consisting of words $x_1, ..., x_n$, the task is to predict a score s_i for every word x_i, such that the words belonging to the ground truth evidence annotated by a human annotator are assigned the highest scores. We perform our experiments in a weakly supervised setting, meaning that no method will have access to samples with actual evidence annotation. Instead, the supervision signal will solely be the label indicating which hypothesis (from a set of 10 possible hypotheses) is addressed in a given abstract. This information will be available during training and testing, as we explore the setting of localizing evidence for a claim that is known in advance.

Our proposed method works by training a standard text classification model to predict the correct hypothesis label for a given sample and to use the MaRC method to extract an explanation for the given label of interest post hoc (see Sect. 4.1 for a detailed description). We hypothesize that this method will outperform other interpretability methods, as it is explicitly designed to generate human-like rationales in the form of consecutive spans of text. As we are not aware of other methods for weakly supervised claim localization, we evaluate this method against other explainability methods (see Appendix A for an overview) as well as against a supervised baseline to allow for a relative performance comparison. For model and training details, see Appendix A.

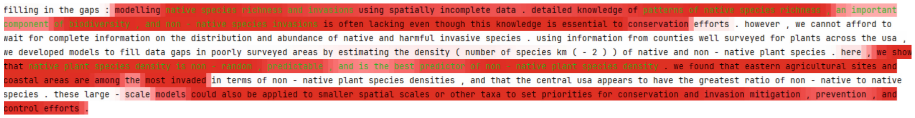
We additionally employ a post-processing step in our prediction pipeline: We split the abstract into individual sentences using ScispaCy [21] and set the predicted scores of the last token of each sentence to 0. This additional step improves span-matching performance, since claim evidence annotations in our particular task do not cross sentence boundaries and do not include punctuation.

Evaluation. We evaluate different measures for the quality of the predicted scores. To assess the quality of the scores assigned to the individual words (independent of their belonging to a longer span of text) we evaluate the area under the precision-recall-curve (*AUC-PR*).

We also evaluate the F1 score, which requires a binary prediction (i.e., each word is either predicted to belong to the evidence or not). Since many methods do not have an obvious way of determining a score threshold, we select the $p \cdot n$ highest-scoring words and average over 19 values of p (0.05, 0.10, 0.15, ..., 0.95).

The same technique is used for the *IoU-F1* score, which we propose as a measure for determining the quality of predicted spans of text. Given a binary prediction for each token, we determine predicted spans as continuous spans of words that were selected as evidence and calculate the IoU between all pairs of predicted and ground truth spans. As perfect matches are unlikely for this challenging task, we define generalized versions of precision and recall that allow for partial matches. To do so, we determine the highest IoU value of each span (ground truth and predicted) with anyone from the other set, and define the precision as the average of these highest values for the ground truth spans, which, analogous to the usual precision, is a measure for how well the ground truth spans have been recognized. Similarly, we define the recall as the average over the highest values for the predicted spans, thus measuring how likely a predicted span matches any of the ground truth spans. The F1 score is calculated from these values as usual and is again averaged over all values of p.

The three scores described so far are well-suited for comparing different methods with each other. To give a better feeling for the absolute quality of the predictions, we again use the F1 and IoU-F1 scores (now denoted as *D-F1* and *D-IoU-F1*), but for a single selection of words: We select a threshold t as the score of the k-th highest-scoring word, with k being the number of words in the ground truth evidence. As ground truth information is used, this is not an objec-



filling in the gaps : modelling native species richness and invasions using spatially incomplete data . detailed knowledge of patterns of native species richness , an important component of biodiversity , and non - native species invasions is often lacking even though this knowledge is essential to conservation efforts . however , we cannot afford to wait for complete information on the distribution and abundance of native and harmful invasive species . using information from counties well surveyed for plants across the usa , we developed models to fill data gaps in poorly surveyed areas by estimating the density (number of species km (- 2)) of native and non - native plant species . here , we show that native plant species density is non - random , predictable , and is the best predictor of non - native plant species density . we found that eastern agricultural sites and coastal areas are among the most invaded in terms of non - native plant species densities , and that the central usa appears to have the greatest ratio of non - native to native species . these large - scale models could also be applied to smaller spatial scales or other taxa to set priorities for conservation and invasion mitigation , prevention , and control efforts .

Fig. 1. Exemplary prediction of the MaRC method for an abstract from the INAS dataset for the *Biotic Resistance Hypothesis* label. Green text marks ground truth annotations, red spans indicate predicted scores.

tive measure of quality, but it nevertheless provides a more interpretable score. We additionally alter the IoU-F1 from the generalized, continuous version to a discrete one used in [6]: A ground truth span is counted as correctly recognized if any predicted span has an IoU of over 0.5, which allows for the calculation of standard precision and recall scores.

5.2 Sentence-Level Claim Localization

Experimental Setup. We perform experiments on weakly-supervised sentence-level evidence localization on the SciFact dataset, which is analogous to the task defined in Sect. 5.1, with the difference that each sentence receives only a single score. Since most explainability methods do not focus on complete sentences, we instead focus on testing different versions of the approach described in Sect. 4.2 and compare them to a supervised baseline, which is a RoBERTa-large classifier [18] that receives a textual claim and a sentence from the abstract and predicts the likelihood of this sentence belonging to the evidence.

We explore different versions of our approach, which differ in the way the base-classifier is trained: As a baseline, we test a classifier that is trained as usual on the SciFact dataset only. We also test a version that is trained with added spans of *PAD* tokens between sentences to align the input spaces present during training and optimization. We also explore the effect of pretraining on five other datasets (Fever [33], EvidenceInference [7,17], PubmedQA [15], HealthVer [26], COVIDFact [25]), which has been shown to improve the classifier performance [37]. Lastly, we also try a supervised version of our approach by employing the procedure described in Sect. 4.3 during classifier training. For more details on the training and evaluation, see Appendix A.

Evaluation. We again evaluate the AUC-PR as a holistic measure of the assigned ranking between the sentences. As for more interpretable measures, we provide the *precision@k* with $k \in \{1, 2, 3\}$, which is defined as the number of ground truth sentences correctly placed among the top-k scoring sentences by the classifier, divided by the maximum number possible (the minimum of the number of available ground truth sentences and k).

For all trained base classifiers, we also provide the F1 score of the abstract-level classification task (*Clf-F1*) to display the effect the different training paradigms have on the classifier performance.

6 Results

6.1 Span-Level Evidence Localization

The results for the span-level evidence localization are displayed in Table 1, while an exemplary output for the MaRC method is displayed in Fig. 1.

The MaRC method outperforms all other methods tested, both for scores measuring token-level performance (AUC-PR, F1, D-F1) as well as for scores evaluating span predictions (IoU-F1, D-IoU-F1). Especially with regards to the span predictions, we see that the MaRC approach significantly outperforms all other methods, which can be explained by it being explicitly designed to produce rationales that mirror human reasoning. The difference to other methods is here, that complete spans are selected as evidence, including words like “the”, “and”, etc., if they are directly part of an important span. Other methods, in comparison, mainly select the few rare words that are a more direct hint towards the hypothesis label, but do therefore not match human-annotated spans. This phenomenon also negatively affects token-level scores for other methods, since only few words per span are recognized as important. For the occlusion method, we produce a similar behavior by occluding longer spans of text at a time, leading to smoothly varying scores and thus to the only method that remotely rivals the MaRC method.

Notably, some methods barely outperform a random baseline (especially for span prediction evaluations), thus making them unusable for claim localization. As a possible explanation, [3] analyzed that classifiers for this task can make use of individual words like species names or locations as hints for the hypothesis if these names only occur in the context of this specific hypothesis. These will not be annotated by the human annotators, though, as hypothesis evidence (according to our definition) needs to clearly reference parts of the respective hypothesis. Overall, this shows a limitation of the proposed approach of using explainability methods for claim localization, as this approach relies on a high overlap between spans considered by humans as hypothesis evidence and words actually used by the classifier as the basis for the prediction, which is not always given.

As is to be expected, though, all methods are outperformed significantly by the supervised baseline. It is the only method that is explicitly trained to predict spans of the desired form, and the only method that has knowledge about the type of information that is to be selected. For weakly supervised methods, that do not have any of this information, predicting the precise span boundaries is extremely difficult. This result suggests, that for a smaller prediction space results could be improved, which we analyzed for the case of sentence-level evidence localization.

6.2 Sentence-Level Evidence Localization

The results for the sentence-level evidence localization are displayed in Table 2. Even though we altered the existing MaRC approach due to the differences between the tasks, our proposed method is still denoted as “MaRC”.

Table 1. Results for the span-level claim localization task on the INAS dataset.

Method	AUC-PR	F1	IoU-F1	D-F1	D-IoU-F1
MaRC	**0.357**	**0.331**	**0.210**	**0.350**	**0.151**
Occlusion	0.310	0.288	0.148	0.310	0.074
Saliency$_{L2}$	0.295	0.311	0.094	0.313	0.019
Saliency$_{Sum}$	0.241	0.265	0.070	0.259	0.002
InXGrad$_{L2}$	0.267	0.304	0.087	0.301	0.013
InXGrad$_{Sum}$	0.240	0.258	0.070	0.248	0.002
Int. Grads$_{L2}$	0.317	0.311	0.091	0.319	0.020
Int. Grads$_{Sum}$	0.320	0.305	0.090	0.322	0.017
LIME	0.271	0.281	0.072	0.273	0.004
Shapley	0.322	0.305	0.086	0.329	0.016
Random	0.221	0.256	0.067	0.223	0.003
Supervised	0.574	0.409	0.231	0.521	0.288

Table 2. Results for the sentence-level claim localization task on the SciFact dataset.

gt	pad	pre	sup	Method	Clf-F1	AUC-PR	Prec@1	Prec@2	Prec@3
X				MaRC	0.859	0.546	0.524	0.578	0.659
				MaRC	0.859	0.581	0.534	0.617	0.741
X	X			MaRC	0.842	0.632	0.612	0.675	0.710
	X			MaRC	0.842	0.655	0.641	0.689	0.736
X	X	X		MaRC	0.877	0.696	0.718	0.738	0.786
	X	X		MaRC	0.877	0.650	0.650	0.699	0.754
X	X	X	X	MaRC	0.936	0.720	0.757	0.772	0.780
	X	X	X	MaRC	0.936	0.718	0.757	0.777	0.780
			X	Sent-clf		0.882	0.883	0.893	0.905
		X	X	Sent-clf		0.902	0.883	0.898	0.951
		X		Sent-clf		0.664	0.650	0.655	0.778

The first four columns in Table 2 provide information about whether the model had access to the ground truth label during optimization (column *gt*), whether the base classifier was trained with added *PAD* tokens (column *pad*), whether the classifier was pretrained (column *pre*) and whether the classifier was trained using evidence supervision (column *sup*).

As, again, no previous study addressed our specific task of weakly supervised claim localization, and since none of the standard explainability methods tested on the INAS dataset proved particularly well-suited for the task at hand, we focus in this section on a comparison of our method with a supervised baseline, and analyze the challenges and solutions for mitigating the gap in performance.

Our most basic version of the MaRC approach (rows 1 and 2) uses a classifier trained without any changes to the standard training procedure. Even for this case, we already see reasonable performance, as it ranks an evidence sentence at the top in 52.4% of cases. Notably, if the optimization is done with respect to the ground truth label (row 1), the performance decreases compared to optimizing with respect to the predicted label (row 2), which is the case for the two lowest-performing base classifiers (up to row 4). This suggests, that without pretraining, the classifier is able to correctly identify the important sentences, but does not have the necessary capabilities to correctly infer the correct label from them.

Our second base classifier (rows 3 and 4) is trained in the same way as before, but receives samples with added *PAD* tokens during training, as these will be common during optimization, leading to otherwise misaligned input spaces. We see a significant improvement for the ground truth and the predicted label cases, so that we train all upcoming classifiers in this way. For this setting, only with access to the weak supervision labels on the SciFact dataset, the MaRC method manages to identify an evidence sentence as the most important sentence in 64.1% of cases, which we already consider quite good performance.

For our next classifier, we added additional pretraining on five similar datasets to the training procedure. This significantly improved the classifier performance and also led to improved results for evidence localization. Notably, from this point onward, having access to the ground truth label during optimization does improve evidence localization performance, indicating that pretraining increased the model's capability of inferring the correct label from the given sentences. Here, we also see the highest performance that we achieved using only weak supervision, with an evidence sentence being correctly identified as most important in 71.8% of cases.

Finally, we experiment with incorporating evidence supervision into the classifier training (as described in Sect. 4.3), to see how far the performance of our method can be pushed in a supervised setting.

At first, we note a significant improvement in the model's general classification performance, which even surpasses the improvement achieved by pretraining. This shows that the evidence injection strategy helped the model with actually understanding the rationale behind specific classifications, which seems to drastically boost the generalization performance.

On the other hand, we also see a significant improvement in the evidence localization results, which could be explained by the better general understanding of the model. We also hypothesize, that this is caused by the general setting of this task: Given an abstract and a claim, the model is supposed to predict one of three labels: *Supports*, *Refutes* or *Not Enough Info*. This means, that sentences that indicate that the general topic of the given abstract aligns with the given claim are considered important (even if they do not directly support or refute the claim) as they affect the likelihood of the *Not Enough Info* label. This leads to these sentences being selected by the MaRC approach as well, as it aims at maximizing the *Supports* or *Refutes* label. Our supervision approach mitigates this behavior, as it explicitly teaches the model to only take actual evidence sentences into account for the classification.

As is to be expected, the supervised baseline models with access to supervision on the SciFact dataset (rows 9 and 10) significantly outperform the weakly supervised models. For a more fair comparison, we also trained a supervised model only on the pretraining datasets and applied it to the SciFact dataset without any supervised training. In this case, the performance of the supervised classifier actually lags behind the MaRC approach in a similar setting (row 5), indicating that, if only abstract-level labels are present, the approach proposed in this work is a valid choice.

In summary, we managed to highlight several problems for our method, ranging from misaligned input spaces and insufficient understanding of the evidence sentences to the selection of non-evidence sentences due to the particular setup of the given task. Many problems can be mitigated by altering the training paradigm of the base classifier, but closing the gap to supervised models still proves to be a significant challenge.

7 Conclusion

In this work, we explored the possibility of using abstract-level labels about the general presence of a claim in this abstract to localize corresponding claim evidence. We proved that this is possible in both the span-level and sentence-level localization settings, but found that the complexity of precise span prediction makes achieving good performance challenging. For the sentence-level task, we found that weakly supervised methods can achieve reasonable performance and even be competitive in settings with only abstract-level labels available.

Since annotating a large number of samples with evidence annotations is very time-intensive and costly, we believe this to be an interesting direction for future research. Especially the fact that evidence supervision during classifier training can improve the performance of explainability methods on this task indicates, that creative changes to the training procedure of neural networks might lead to a substantial improvement of weakly supervised methods, which provides interesting possibilities for future research.

A Experimental Details

Model Details. We use PubMedBERT [12] and RoBERTa large [18] as the classification models for the INAS dataset and SciFact dataset, respectively. We train seven models, and keep the three best performing models with the highest validation F1 score.

The pretraining for the SciFact model is done on five datasets: Fever [33], EvidenceInference [7,17], PubmedQA [15], HealthVer [26], COVIDFact [25]. **MaRC Details.** The optimization for the MaRC method is done with respect to all three trained models. The parameters are set as described in [4], but we employ a new sparsity regularizer that actively forces a maximum average mask value. Similar to [4], we use the following weight regularizer:

$$\varOmega_\lambda = \alpha_\lambda \left[\frac{1}{n} \sum_{i=1}^{n} \lambda_i \right]^2$$

but dynamically update α_λ at each gradient descent step i using to following formulas, to reach a maximum average mask value t (set to 0.35):

$$m_i = \frac{1}{n} \sum_{i=1}^{n} \lambda_i$$

$$\varDelta_i = m_{i-1} - m_i$$

$$\varDelta_{target} = (m_i - t)/150$$

Here, m_i is the current mask mean, $\varDelta_i$ is the difference in mask means from the current optimization step to the last, and $\varDelta_{target}$ is the desired value for $\varDelta_i$, which (if it is always optimal) ensures a steady but decelerating trajectory towards the optimal mask value. We define

$$\varDelta_{\varDelta_i, target} = \varDelta_i - \varDelta_{target}$$

to be the difference between our current single-step mask mean difference and the desired one, which we want to bring as close to 0 as possible. We then define our update for α_λ at iteration i as follows:

$$\alpha_\lambda^i = \alpha_\lambda^{i-1} \cdot (0.8 + 0.2 \cdot \gamma)$$

$$\gamma = \max\left(0.7, 1 - 0.9 \cdot \tanh\left(\frac{1}{0.002}\left(\frac{\varDelta_{\varDelta_i, target}}{2} - (\varDelta_{i-1} - \varDelta_i)\right)\right)\right)$$

so that $\gamma > 1$ leads to an increase in α_λ whereas $\gamma < 1$ leads to a decrease. The max operator prevents an overly steep decrease of α_λ, while the tanh is used to keep positive updates limited. The updates are mainly determined by $\varDelta_{\varDelta_i, target}$, so that α_λ increases when $\varDelta_i$ is smaller than $\varDelta_{target}$ and vice versa. The term $(\varDelta_{i-1} - \varDelta_i)$ is a second-order statistic to prevent "overshooting" in the form of changing α_λ further if $\varDelta_i$ is already approaching $\varDelta_{target}$ (which might take a while due to the momentum-based optimizer).

To give the optimization process the freedom to determine the optimal average mask value on its own after falling below $t + 0.1$, we alter the process of determining α_λ in the following way:

$$\alpha_\lambda^i = \alpha_\lambda^{i-1} \cdot (0.8 + 0.2 \cdot \gamma_{pred} \cdot \gamma_{weight})$$

$$\gamma_{pred} = \min\left(\frac{\mathcal{L}(\tilde{x}, c)_i}{\mathcal{L}(x, c)_0}, 0.5 \cdot \frac{\mathcal{L}(x, c)_0}{\mathcal{L}(\tilde{x}^{\mathsf{c}}, c)_i}, 1.1\right)$$

$$\gamma_{weight} = 1 + (m_i - 0.3)$$

Here, γ_{pred} pushes mask values further down if the model prediction for the current masked input is more confident than the initial unmasked prediction and the prediction for complement mask input is sufficiently less confident than the initial unmasked prediction, which indicates that more information can be removed. γ_{weight} pushes the average mask value to a value of 0.3, since values far below that lead to most words having scores close to 0, and thus to no clear ranking existing among them.

Comparison Methods. The other explainability methods are all used for each of the three models individually, and the scores are averaged afterward. We make use of the following methods and hyperparameter settings:

- *Occlusion* [39]: We chose to mask slightly larger spans of 5 tokens as this produced smoother masks which resulted in higher IoU F1 scores. Occluded parts were replaced by *PAD*-tokens.
- *Saliency* [28]: No hyperparameter settings required.
- *InXGrad* (Input times gradient [27]): No hyperparameter settings required.
- *Int. Grads* (Integrated Gradients [30]): We use a sequence of *PAD*-tokens as background and do 50 gradient evaluation steps per sample.
- *LIME* [24]: We do 50 function evaluations per sample. In each evaluation, we randomly select $5 - 13\%$ of tokens and replace them as well as the next three tokens with *PAD*-tokens. We train a linear classifier and use the resulting weights as rationale.
- *Shapley* (Shapley value sampling [5]): We evaluate the token contributions for 15 feature permutations per sample. Removed tokens are replaced by *PAD*-tokens.

We use the implementations provided by [16] for all methods. All methods have access to the ground truth label. The InXGrad, Saliency and Int. Grads methods all predict one score for each element of the embedding vector of a given word, which is reduced to a single score by using the L2-norm or the sum.

We also compare against a supervised baseline. It is trained on 517 samples from the INAS dataset annotated with span-level evidence, as well as on 204 samples without annotated evidence. To make use of the samples without evidence annotations we train in a multi-task setting by also training to predict the general hypothesis labels for the whole abstract.

INAS Evaluation. We evaluate all methods on a test set consisting of 141 samples that cover all ten possible classes. The test set contains all 50 samples that were annotated by all three annotators, as well as 91 further samples that were annotated by only one of the three annotators, with samples and annotators being assigned randomly. For the samples that were annotated by all annotators, we create a single ground truth by taking the intersection of the set of annotated tokens between each pair of annotators, followed by the union between the three resulting annotations of each pair.

SciFact Evaluation. As the test labels for the SciFact dataset are not publicly available, we create new splits with 50 claims for validation, 150 claims for testing and the remaining claims for training. The actual samples for the splits can then be created from the given claims and linked documents.

For evaluating the *AUC-PR* and *Precision@k* scores, we only take samples from the *Supports* and *Refutes* classes into account, as they are the only classes with corresponding evidence annotations.

References

1. Accuosto, P., Neves, M.L., Saggion, H.: Argumentation mining in scientific literature: from computational linguistics to biomedicine. In: BIR@ECIR (2021)
2. Blake, C.: Beyond genes, proteins, and abstracts: identifying scientific claims from full-text biomedical articles. J. Biomed. Inform. **43**(2), 173–189 (2010). https://doi.org/10.1016/j.jbi.2009.11.001
3. Brinner, M., Heger, T., Zarriess, S.: Linking a hypothesis network from the domain of invasion biology to a corpus of scientific abstracts: the INAS dataset. In: Proceedings of the first Workshop on Information Extraction from Scientific Publications, pp. 32–42. Association for Computational Linguistics (2022)
4. Brinner, M., Zarrieß, S.: Model interpretability and rationale extraction by input mask optimization. In: Findings of the Association for Computational Linguistics: ACL 2023, pp. 13722–13744. Association for Computational Linguistics (2023). https://doi.org/10.18653/v1/2023.findings-acl.867
5. Castro, J., Gómez, D., Tejada, J.: Polynomial calculation of the Shapley value based on sampling. Comput. Oper. Res. **36**(5), 1726–1730 (2009). https://doi.org/10.1016/j.cor.2008.04.004
6. DeYoung, J., Jain, S., Rajani, N.F., Lehman, E., Xiong, C., Socher, R., Wallace, B.C.: ERASER: a benchmark to evaluate rationalized NLP models. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 4443–4458. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.acl-main.408
7. DeYoung, J., Lehman, E.P., Nye, B.E., Marshall, I.J., Wallace, B.C.: Evidence inference 2.0: more data, better models. arXiv abs/2005.04177 (2020)
8. Freeman, J.B.: Dialectics and the macrostructure of arguments. De Gruyter Mouton (1991). https://doi.org/10.1515/9783110875843
9. Fu, M.C.: Chapter 19 gradient estimation. In: Simulation, Handbooks in Operations Research and Management Science, vol. 13, pp. 575–616. Elsevier (2006). https://doi.org/10.1016/S0927-0507(06)13019-4
10. Green, N.: Towards creation of a corpus for argumentation mining the biomedical genetics research literature. In: Proceedings of the First Workshop on Argumentation Mining, pp. 11–18. Association for Computational Linguistics (2014). https://doi.org/10.3115/v1/W14-2102
11. Green, N.: Identifying argumentation schemes in genetics research articles. In: Proceedings of the 2nd Workshop on Argumentation Mining, pp. 12–21. Association for Computational Linguistics (2015). https://doi.org/10.3115/v1/W15-0502
12. Gu, Y., et al.: Domain-specific language model pretraining for biomedical natural language processing. ACM Trans. Comput. Healthc. **3**(1), 1–23 (2022). https://doi.org/10.1145/3458754, arXiv:2007.15779 [cs]

13. Jansen, T., Kuhn, T.: Extracting core claims from scientific articles. In: Bosse, T., Bredeweg, B. (eds.) BNAIC 2016. CCIS, vol. 765, pp. 32–46. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67468-1_3
14. Jeschke, J.M., Heger, T.: Invasion biology: hypotheses and evidence (2018). https://doi.org/10.1079/9781780647647.0000
15. Jin, Q., Dhingra, B., Liu, Z., Cohen, W., Lu, X.: PubMedQA: a dataset for biomedical research question answering. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 2567–2577. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/D19-1259
16. Kokhlikyan, N., et al.: Captum: a unified and generic model interpretability library for PyTorch (2020)
17. Lehman, E., DeYoung, J., Barzilay, R., Wallace, B.C.: Inferring which medical treatments work from reports of clinical trials. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 3705–3717. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/N19-1371
18. Liu, Y., et al..: RoBERTa: a robustly optimized BERT pretraining approach (2019)
19. Lloyd, E.A.: Confirmation of ecological and evolutionary models. Biol. Philos. **2**(3), 277–293 (1987). https://doi.org/10.1007/BF00128834
20. Madsen, A., Reddy, S., Chandar, S.: Post-hoc interpretability for neural NLP: a survey. ACM Comput. Surv. **55**(8) (2022). https://doi.org/10.1145/3546577
21. Neumann, M., King, D., Beltagy, I., Ammar, W.: ScispaCy: fast and robust models for biomedical natural language processing. In: Proceedings of the 18th BioNLP Workshop and Shared Task, pp. 319–327. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/W19-5034
22. Peldszus, A., Stede, M.: From argument diagrams to argumentation mining in texts: a survey. Int. J. Cogn. Inform. Nat. Intell. **7**(1), 1–31 (2013). https://doi.org/10.4018/jcini.2013010101
23. Pradeep, R., Ma, X., Nogueira, R., Lin, J.: Scientific claim verification with VerT5erini. In: Proceedings of the 12th International Workshop on Health Text Mining and Information Analysis, pp. 94–103. Association for Computational Linguistics (2021)
24. Ribeiro, M.T., Singh, S., Guestrin, C.: "why should i trust you?": explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. KDD '16, pp. 1135–1144. Association for Computing Machinery (2016). https://doi.org/10.1145/2939672.2939778
25. Saakyan, A., Chakrabarty, T., Muresan, S.: COVID-fact: fact extraction and verification of real-world claims on COVID-19 pandemic. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pp. 2116–2129. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.acl-long.165
26. Sarrouti, M., Ben Abacha, A., Mrabet, Y., Demner-Fushman, D.: Evidence-based fact-checking of health-related claims. In: Findings of the Association for Computational Linguistics: EMNLP 2021, pp. 3499–3512. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.findings-emnlp.297

27. Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: Proceedings of the 34th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 70, pp. 3145–3153. PMLR (2017)
28. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: visualising image classification models and saliency maps. CoRR (2013)
29. Stab, C., Gurevych, I.: Parsing argumentation structures in persuasive essays. Comput. Linguist. **43**(3), 619–659 (2017). https://doi.org/10.1162/COLI_a_00295
30. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: Proceedings of the 34th International Conference on Machine Learning - Volume 70. ICML'17, pp. 3319–3328. JMLR.org (2017)
31. Teufel, S., Carletta, J., Moens, M.: An annotation scheme for discourse-level argumentation in research articles. In: Ninth Conference of the European Chapter of the Association for Computational Linguistics, pp. 110–117. Association for Computational Linguistics (1999)
32. Teufel, S., Siddharthan, A., Batchelor, C.: Towards domain-independent argumentative zoning: evidence from chemistry and computational linguistics. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pp. 1493–1502. Association for Computational Linguistics (2009)
33. Thorne, J., Vlachos, A., Christodoulopoulos, C., Mittal, A.: FEVER: a large-scale dataset for fact extraction and VERification. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 809–819. Association for Computational Linguistics (2018). https://doi.org/10.18653/v1/N18-1074
34. Toulmin, S.: The Uses of Argument. Cambridge University Press, Cambridge (1958)
35. Wadden, D., et al.: Fact or fiction: verifying scientific claims. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 7534–7550. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.emnlp-main.609
36. Wadden, D., et al.: SciFact-open: towards open-domain scientific claim verification. In: Findings of the Association for Computational Linguistics: EMNLP 2022, pp. 4719–4734. Association for Computational Linguistics (2022). https://doi.org/10.18653/v1/2022.findings-emnlp.347
37. Wadden, D., Lo, K., Wang, L.L., Cohan, A., Beltagy, I., Hajishirzi, H.: MultiVerS: improving scientific claim verification with weak supervision and full-document context. In: Findings of the Association for Computational Linguistics: NAACL 2022, pp. 61–76. Association for Computational Linguistics (2022). https://doi.org/10.18653/v1/2022.findings-naacl.6
38. Zaidan, O., Eisner, J., Piatko, C.: Using "annotator rationales" to improve machine learning for text categorization. In: Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference, pp. 260–267. Association for Computational Linguistics (2007)
39. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_53

40. Zeng, X., Abumansour, A.S., Zubiaga, A.: Automated fact-checking: a survey. Lang. Linguist. Compass **15**(10), e12438 (2021). https://doi.org/10.1111/lnc3.12438

41. Zhang, Z., Li, J., Fukumoto, F., Ye, Y.: Abstract, rationale, stance: a joint model for scientific claim verification. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 3580–3586. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.emnlp-main.290

Argument Mining of Attack and Support Patterns in Dialogical Conversations with Sequential Pattern Mining

Mattes Ruckdeschel[1](✉), Ringo Baumann[2], and Gregor Wiedemann[1]

[1] Leibniz-Institute for Media Research | Hans-Bredow-Institut, Rothenbaumchaussee 36, 20148 Hamburg, Germany
m.ruckdeschel@leibniz-hbi.de
[2] Universität Leipzig, Augustusplatz 10, 04109 Leipzig, Germany

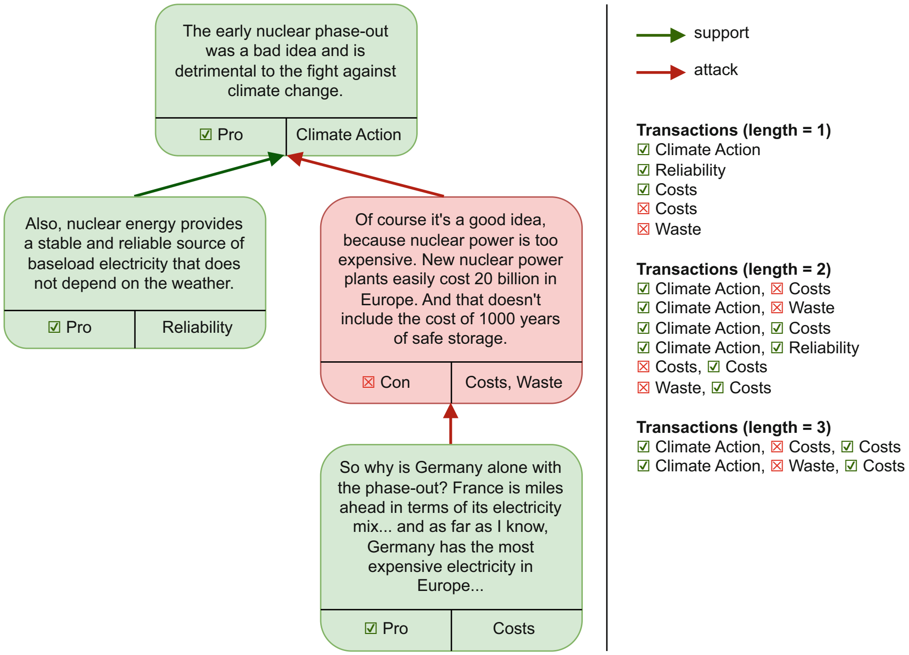
Abstract. Argument mining usually operates on short, decontextualized argumentative units such as main and subordinate clauses, or full sentences as proxies for arguments. Argumentation in digital media environments, however, is embedded in larger contexts. Especially on social media platforms, argumentation unfolds in dialog threads or tree structures where users interact with each other. To reveal patterns of such interactions, we transform 2.5 million tweets from 38k German Twitter conversations concerning nuclear energy from 2017, 2019, and 2021 into an abstract representation encoding their stance, and aspects. We then apply Sequential Pattern Mining, a common method for finding patterns in large databases, and explore its capabilities to investigate typical argumentation schemes in user debates. The approach reveals distinct patterns of support and attack relations between pro and contra arguments about nuclear energy in conversational threads when comparing different time slices of our corpus. For example, we are seeing an increasing relevance of the climate aspect in attacks on anti-nuclear arguments. However, the pro arguments are increasingly being countered by cost aspects. Analyzing this diachronic change of patterns allows us to describe the discursive processes of argumentation on a macro level that drive the slow but steady transformation of a society's social and political convictions.

Keywords: Pattern Mining · Computational Social Science · Aspect-Based Argument Mining

1 Mining Interactions in Debates

In recent years, there have been many refinements to argument mining (AM), a natural language processing (NLP) sub-task that deals with the detection and classification of argumentative structures in text [12]. With the advent of modern transformer-based language models such as BERT [7] and its numerous successors, text classification of increasingly abstract categories such as frames [9] or

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 39–56, 2024.
https://doi.org/10.1007/978-3-031-63536-6_3



The early nuclear phase-out was a bad idea and is detrimental to the fight against climate change.

Also, nuclear energy provides a stable and reliable source of baseload electricity that does not depend on the weather.

Of course it's a good idea, because nuclear power is too expensive. New nuclear power plants easily cost 20 billion in Europe. And that doesn't include the cost of 1000 years of safe storage.

So why is Germany alone with the phase-out? France is miles ahead in terms of its electricity mix... and as far as I know, Germany has the most expensive electricity in Europe...

Transactions (length = 1)
☑ Climate Action
☑ Reliability
☑ Costs
☒ Costs
☒ Waste

Transactions (length = 2)
☑ Climate Action, ☒ Costs
☑ Climate Action, ☒ Waste
☑ Climate Action, ☑ Costs
☑ Climate Action, ☑ Reliability
☒ Costs, ☑ Costs
☒ Waste, ☑ Costs

Transactions (length = 3)
☑ Climate Action, ☒ Costs, ☑ Costs
☑ Climate Action, ☒ Waste, ☑ Costs

Fig. 1. Example of a tree structure of a Twitter conversation about nuclear energy. Vertices are Tweets, directed edges indicate replies. For SPM, argumentative tweets are converted to abstract representations encoding stance (pro, or contra) and aspects (one or more labels from a set of 17 aspects describing the German nuclear energy debate, cf. Table 3). For each tree, a set of transactions of certain lengths for pattern mining can be derived.

aspects [18] have been introduced. While these new methods are improving our abilities to capture argument semantics, they still operate with isolated text units that only approximate the kind of argumentation that occurs in the wild. Among other things, this can be attributed to the characteristics of the fine-tuning of language models for text classification: the process operates with limited context lengths and works best with an abundance of singularly labeled data points. In real life, however, argumentation regularly takes place as an exchange of arguments. An abundance of those exchanges can nowadays be observed on online social media platforms, ready to be analyzed. This leads to large datasets of structured conversations, rich in potential arguments. While interesting findings can already be found by classifying isolated text units, the information from dialog structures proves valuable for analyzing argumentative discourses more closely.

In this paper, we explore a novel approach to combine state-of-the-art *argument mining* approaches with *sequence pattern mining* (SPM), a data mining approach that is more prominently used in a market research context to answer the following research question: *How can categorical predictions from text classi-*

fication be evaluated together with dialogical structural information to find characteristic argumentative patterns that describe the dynamics of a debate?

To create abstract representations of arguments, we fine-tune language models on two common classification tasks for arguments: stance, and aspect. We then apply both classifiers to a large corpus of tweets related to the nuclear energy debate (cf. Figure 1 for an example graph of an original tweet and its replies). After mining this dataset for reply chains of various lengths, we can describe interactions between users as sequences of tuples in the form (*aspect*, *stance*), and look for common patterns in this database. We can then further examine the conversations that contain the most frequent patterns qualitatively and see if they allow us to draw conclusions about how people react to different arguments in online debates. With our method, we aim to support social science research to conduct discourse analysis of large diachronic datasets that utilize the technological advances in NLP constructively.

In the upcoming Sect. 2, we give an overview of related work to our approach. In Sect. 3, we describe the dataset that we have used for conducting our experiments as well as the details regarding the fine-tuned language models that were used. We also introduce our approach to finding patterns in conversations. In Sect. 4, we compare the patterns found in our dataset across the different time slices and conclude in Sect. 5 with a discussion of the potentials as well as the limitations of our approach as a method for argument mining in the social sciences.

2 Related Work

Argument mining advanced to the extraction of finer-grained, more qualitative *features from argumentative text.* Examples include argument mining with a novel focus on key points [8], frames [2] or aspects [18,24], which aim to extend argument mining originally focusing on linguistic structures to more semantic units that are of interest for (computational) social science research. Analyzing semantic aspects of arguments is still not widespread in argument mining due to its challenges to cover the broad range of controversial topics [3], but there is already a solid foundation of preliminary work. [15] stressed the importance of context when mining for argument relations, albeit prior to the advancements of powerful contextual word embeddings. [22] established the task of mining for argumentation structures as an important link to discourse analysis. Newer approaches also include larger contexts to better comply with argumentation patterns in empirical data that often use implicit premises, lack argument markers, or are elaborated beyond single sentences [17]. Widening the context for text classification also proved helpful for other text classification tasks such as hate speech detection [28].

While most work on argument mining focuses on learning from isolated textual units, some research tries to mine argumentation from *dialogue structures* such as online discussion threads [6]. [20] identify distinguishable conversation types from Twitter conversations that can potentially be exploited for mining argument relations. They similarly mined for conversations on Twitter, but have built a smaller dataset by only considering the longest possible thread from an initial root tweet to one leaf. We build upon this work by utilizing the structural information that is available to a greater extent, and include dialogue structures from the many incomplete conversations on Twitter, too.

Moreover, there has been increased attention on the importance of interdisciplinary approaches to argument mining [25]. The field of *computational social science* (CSS) strives to analyze large amounts of digital trace data with computational methods for social science research questions. Argument mining bears a high potential for CSS due to its ability to give insights into the use of argumentation in political, or otherwise socially impactful debates. In recent years, more cooperation between researchers with a strong foundation in both argument mining and CSS was established. These works often produce data annotated in a way that is in line with the existing standards from the social sciences and make supervised machine learning applicable, e.g. [11], and [18]. Such datasets are important for bringing argument mining closer to CSS researchers as they enable thorough quantitative research opportunities and give a new dimension to qualitative research on big data. We build on the work of [18], by using the methodology of annotating data in tandem with experts from social science to create a dataset with high utility. [10] describe a methodology of using Discourse Network Analysis, a network representation obtained from news corpora, where actors (e.g. politicians) and their claims form two types of nodes in a bipartite graph. By this, discourse networks combine state-of-the-art AM technology for claim and stance detection with a social science goal. Our approach differs from this method by relying on explicit dialog structures from empirical conversation data instead of modeling abstract discourse representations from large amounts of news. [14] describe a novel method for predicting argument persuasiveness from *patterns* of types of argumentative discourse units mined from individual posts in online debates which are then clustered with other patterns from the same discourse. The features used are more structural and context-independent and patterns are clustered in order to get insights into discussion. While their approach has a similar goal to ours, namely finding patterns in discussions, it does not employ data mining on patterns but uses clustering of similar sequences on the level of single posts.

Sequential Pattern mining is not widely used today, neither in CSS nor in NLP applications. [26] used SPM for retrieving questions from text in the absence of common cues like question marks, which is common for online utterances that may lack the usual grammatical structure. [21] applied SPM to analyze argument structures for two scientific domains for which they hand-coded argumentative structures. They annotated argumentative sections in scientific articles and used SPM to identify typical argument structure models based on the patterns they found. To our knowledge, our study is the first that utilizes semantic features from argument mining as input for SPM.

Table 1. Dataset statistics of the tweet dataset.

	2017	2019	2021	**Total**
Number of tweets	4869	58984	171098	**234951**
Number of conversations	645	4699	24014	**29358**

3 Predicting a Conversational Dataset

Since SPM operates on ordered sets of items, we need to convert the information of individual utterances of a conversation into elements of sets, creating *transactions* that represent the conversation. We first created a structured conversation corpus that contains conversation trees. A conversation tree is a directed tree graph with tweets as its nodes, and their reply relationship to a previously posted tweet as edges. An example of a conversation tree is shown in Fig. 1. We can then mine the tree structures for *conversation chains* of various length n, which are sub-graphs of the conversation tree. By classifying each node in a chain with the two properties of stance and aspect, we encode arguments in tweets as transactions. We perform pattern mining on the ordered sets of these transactions.

3.1 Corpus Creation

In order to create a dataset of conversations that are held on social media, we mined entire conversations from Twitter (now re-branded as X). For our study, we focus on the nuclear energy debate in Germany. We first used a key term query to the Twitter API to retrieve individual tweets related to the nuclear energy debate in German language from three different years: 2017, 2019, and 2021.[1] The resulting tweets were used to retrieve thematically matching conversations by two strategies. First, we filtered for *root tweets* only, i.e. keyword-matching tweets that were posted on Twitter initially, in contrast to replies as reactions to earlier posted tweets, and requested their entire set of replies via the API. Second, for reply-tweets that matched our query within a conversation, we included these tweets along with their directly connected replies from the conversation tree. While this proceeding reduced the size of our dataset significantly, it was necessary to ensure that the dataset remained consistent with our target topic.

Table 1 shows basic statistics of the final dataset, which is heavily skewed toward the more recent conversations from 2021. This is likely due to an increase in public attention to the topic of nuclear energy as well as the growing popularity of Twitter as a public debate forum. Further, the more conversations date back

[1] The list of key terms comprised *Kernenergie, Atomenergie, Nuklearenergie, Atomkraft, Kernkraft, Atomausstieg*, and *Atomverzicht* and their inflected forms. The time slices were selected, on the one hand, with the requirement to cover a larger period to capture long-term evolvements of the debate. On the other hand, the selection should guarantee substantial dataset sizes for statistical analysis which were only available from the year 2017 onward.

Table 2. Chain length distribution by year

Length	2017	2019	2021
2	691 (41.2%)	9586 (42.5%)	22948 (39.1%)
3	197 (11.8%)	4027 (17.8%)	10632 (18.1%)
4	165 (9.8%)	2265 (10%)	6585 (11.2%)
5	106 (6.3%)	1508 (6.7%)	4379 (7.5%)
6–10	277 (16.5%)	3607 (16%)	9807 (16.7%)
11+	240 (14.3%)	1569 (7%)	4314 (7.4%)
Total	**1676**	**22562**	**58665**

in time, the more likely it is that parts or the entire conversation, were deleted from the platform and, thus, are no longer available via the API.[2]

3.2 Mining Conversation Chains from Incomplete Graphs

Since many of the conversation trees in our dataset referenced tweets that could not be retrieved by the API anymore, we opted for mining chains for each tweet individually as an alternative to the traversal of complete conversation trees. This ensures that all tweets that are included in any chain also have immediate neighbors included in the chain, making the mining of relations between utterances and their responses possible. Table 2 shows the distribution of the reconstructed maximum chain lengths for each year. Around 40% of all tweets in the corpus that are predecessors in a dialogical conversation triggered one single reply only. The longest reply chains we found contain up to 70 messages. We decided to limit chain lengths in our experiments for several reasons. First, computational complexity increases significantly for longer chains. Second, from the low ratio of extremely long chains, it is already evident that the likelihood of finding common argumentative patterns that include a larger number of items will be very low.

3.3 Argument Abstraction by Stance and Aspect Prediction

We aim to use established AM methods to derive tuples of information that represent an abstract version of an argumentative text. Two major semantic pieces of information of an argument are stance and aspect, which can be classified with satisfactory performance by fine-tuned transformer language models on labeled examples. For this, we annotated a dataset of 642 German tweets with their stance and aspect data following the method described in [18]. Table 3 shows

[2] The fact of incomplete conversations makes research on historic data more challenging. Overall, only around 27% of all retrieved conversations contained a complete conversation tree. our chain mining procedure addresses this problem by focusing on sub-trees around matching key terms.

the aspects that were coded to cover the most prominent aspects of the German debate. Intercoder-agreement measured by Krippendorff's α yields very good agreement for most of the categories. Two aspect categories, temporal dimension, and reliability, achieved only substantial agreement around 0.6.[3] In addition, we used large publicly available English-language datasets on both tasks for transfer learning in a multitask learning (MTL) setting. As a language model, we used the multilingual version `xlm-roberta-large` of RoBERTa [13] in all our experiments. Further, for all experiments, five models were trained to minimize random effects in the results. We report the mean performance and standard deviation of the performance in Table 4. For aspect classification, we used all available data from the Argument Aspect Corpus (AAC) [19] for transfer learning, which contains aspect labels for sentences from four topics, and our additionally coded German-language dataset in a two-task MTL sequence tagging. On the test set of 10% of the annotated German tweets, our classifier achieved an overall micro F1-score of 77%.[4] For stance classification, we used the Sentential Argument Mining Corpus (UKP-SAM) [23], which provides stance information on a large number of sentences across eight topics, as a transfer learning task. We modeled both tasks, the UKP-SAM dataset and the additional German tweet dataset, as text classification tasks. The classifier reaches a micro F1-score of around 80% on the German test data.

3.4 Sequential Pattern Mining on Predicted Data

SPM aims to find reoccurring patterns in databases containing sequentially ordered transactions [1]. The method is typically employed to identify patterns for market basket analysis such as 'customers who bought a PC, and later that month a digital camera likely will buy a printer next month'. For our analysis, we conceptualize dialogical argumentation threads analogous to shopping cart analysis as compilations of abstract augmentations from the 'market' of publicly debated ideas. We build transactions by representing each tweet in a retrieved chain with a tuple representation containing the predicted aspect and stance information. We use the *PrefixSpan* algorithm [16][5], which efficiently finds patterns by recursively building from their prefixes, starting with all prefixes of length 1. In each step, for each prefix α, the *projected database* $S|_{\alpha}$ of α is created, which contains all *postfixes* of α, which are all sub-patterns that start with α.[6] The most important metric for evaluating the significance of mined sequences is the *support*, which is defined as the proportion of the number of sequences in which a pattern occurs. As a parameter, PrefixSpan considers in each step only postfixes with a minimum desired support. After some experimental testing on

[3] One category, public opinion, was discarded from the dataset as it did not achieve substantial agreement.

[4] In accordance with [18], we evaluate aspect tagging on the tweet level, since we were interested in the aspects related to an entire tweet instead of its specific tokens.

[5] We used the implementation from https://github.com/chuanconggao/PrefixSpan-py.

[6] A comprehensible example can be found in [27].

Table 3. Number of occurrences and intercoder agreement (Krippendorff's α) for each aspect in the tweet dataset. In the paper, we refer to aspects using the corresponding English short labels.

Category	English short label	N	α_K
Abfall/Atommüll	waste	65	0.91
Autonomie/Abhängigkeit	autonomy	50	0.82
Erneuerbare Energien	renew(ables)	143	0.90
Fossile Brennstoffe	fossil fuels	141	0.88
Gesundheitliche Auswirkungen	health	39	0.87
Klimaschutz	climate	133	0.83
Kosten	costs	117	0.76
Lobbyismus	lobbyism	18	0.79
Nachhaligkeit	sustainability	33	0.72
Sicherheit und Unfälle	safety	127	0.78
Technologische Innovation	innovation	61	0.80
Umweltschutz	environment	47	0.79
Waffen	weapons	12	0.83
Wissenschaftlichkeit	science	55	0.81
Zeitliche Dimension	temporality	108	0.59
Zuverlässigkeit	reliability	109	0.61
All Topics	–	**1303**	**0.63** -

our empirical data, we set the minimum support for patterns considered relevant for our analysis to 1%.[7]

Figure 2 shows the stance distribution for the predicted dataset. A significant proportion of the tweets in the dataset were predicted as having no stance. This is plausible since not all posts for a topic are actually argumentative and pose a stance. For the pattern mining experiments, chains that contained tweets

Table 4. Overall performance metrics for sentence-level aspect classification and stance classification on the test dataset of coded tweets concerning the German nuclear energy debate.

	Precision	Recall	F1-score
Sentence-level aspect classification	0.76 ± 0.03	0.77 ± 0.01	0.77 ± 0.01
Stance classification	0.80 ± 0.03	0.80 ± 0.03	0.80 ± 0.03

[7] PrefixSpan does, however, consider non-contiguous patterns, i.e. (a, c) may be a frequent pattern of (a, b, c). While we consider this potentially problematic for attack and support pattern mining in general databases, our database, consisting of predominately short patterns should still yield sufficiently relevant patterns.

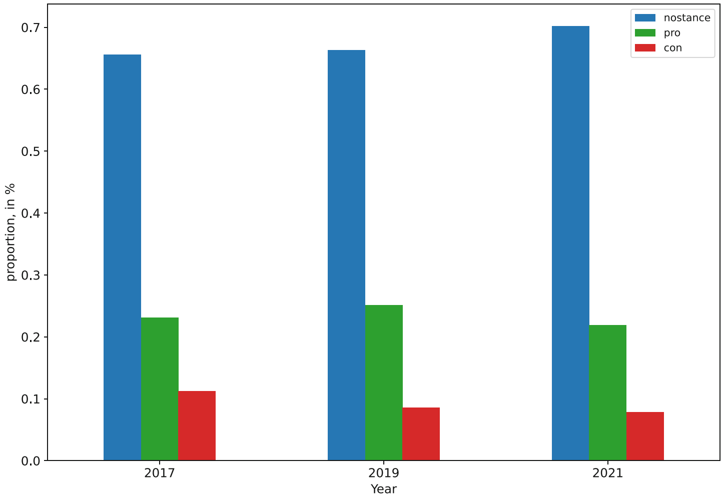


Fig. 2. Stance distribution in the predicted dataset, by year

without a stance were excluded. This was due to the fact that including these posts resulted in a majority of patterns revolving around tweets without a stance, which were not argumentative, thus revealing no argumentative patterns. It is also noticeable that a majority of tweets with a stance were predicted as having a *pro* stance. While in 2017 there are 2.05 times more pro tweets than con tweets, this factor increases by almost 50% to 2.94 in 2019 and slightly decreases to 2.79 in 2021. This implies that the discussion on Twitter is generally more in favor of nuclear energy.

Since we tagged aspects as token spans, one tweet can potentially contain multiple aspects. We investigated two possibilities to resolve multi-aspect tweets to create transactions. First, *concatenation* of aspects, e.g. `(costs_reliability, pro)` for a tweet with a pro stance which contains costs and reliability as aspects. Alternatively, we create *flat* representations, creating separate transactions for each aspect. We found that concatenating aspects resulted in fewer significant patterns, as a result of the combinatorial explosion of possible transactions (see Fig. 5 in the Appendix for a discussion of this processing step). Due to these two findings, we limit the mining for patterns on flat chains to chains that contain only tweets for which a pro or con stance was predicted.

4 Results

Figures 3 and 4 show the proportions of aspects for pro and con stanced tweets, i.e. patterns of length 1, by year. For pro arguments, three aspects have a share of more than 10% of all pro arguments throughout the three years: *renewables*, *fossil fuels*, and *climate*. Two other aspects, *safety* and *reliability* fall below 10%

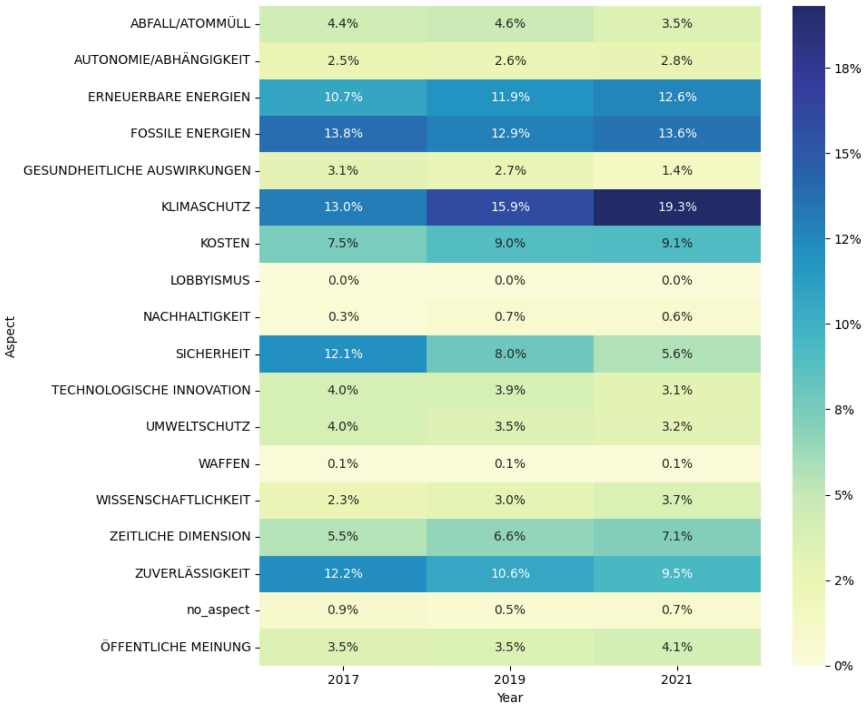


Fig. 3. Aspect distribution for pro arguments, by year

of shares, and other aspects generally make up five or less percent of all pro arguments. The most significant increase is seen in the share of arguments addressing *renewables*, which make up nearly 20% in 2021. For con arguments, *renewables*, *costs* and *safety* are strongly represented throughout the years, but a greater number of aspects are represented between five and ten percent throughout the years. While *climate* is steadily rising from six to ten percent, *reliability* is falling to 8.3%. An important difference between the two distributions is the prevalence of *nuclear waste* as a well-represented con argument while staying below a five percent proportion throughout the years in contexts of a pro argument.

4.1 Attack and Support Patterns

Table 5 shows the top five patterns for the four possible combinations of pro and con-stanced tweets over the three time slices. Alteration between pro and con stances in subsequent tweets of a chain can be interpreted as an attack relation of arguments while repeated stances indicate a support relation. The most significant patterns all have a length of two. In total, 327 patterns with minimum support of 1% were mined, yet only 24 patterns had more than two items. Due to the chain length distribution in the dataset (cf. Table 2), longer

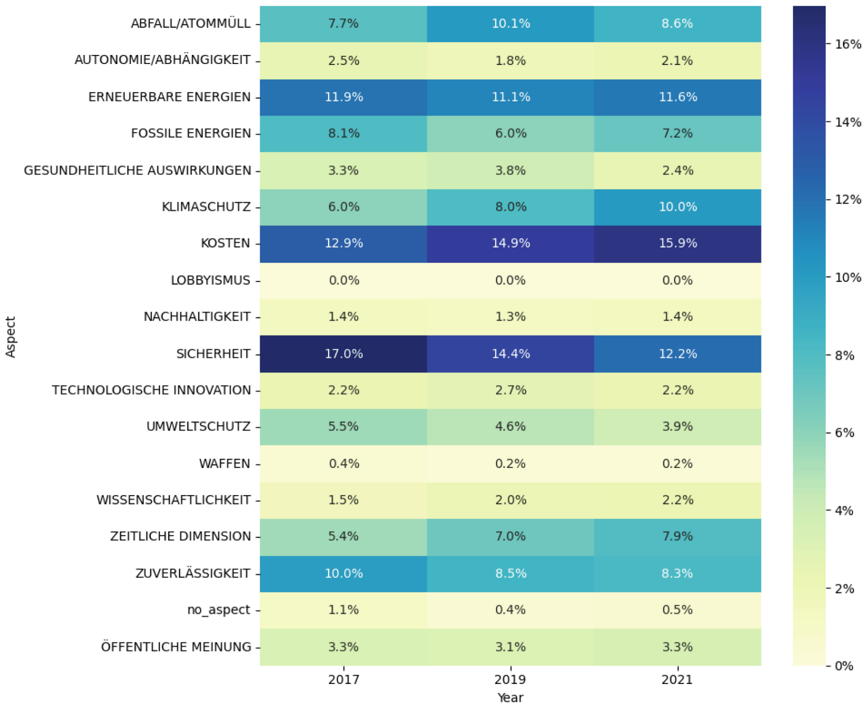


Fig. 4. Aspect distribution for contra arguments, by year

chains are hardly found in the dataset. The support of the top patterns of 2017 is significantly higher compared to later years. A possible explanation is that the smaller overall discourse by number of tweets was more uniform and expanded over time to more diverse aspects. We further observe the highest support for *pro* ← *pro* patterns, which originates from the high prevalence of pro-labeled tweets in the dataset. Many prevalent patterns address the same aspect in a row. A possible explanation is that people prefer to reinforce statements they agree with by repeating them (with variations). Another factor may be self-replies to construct a longer thread of tweets for making an argument. In the following, we investigate the results for each combination of pro and con-stanced tweets.

Support Pro ← Pro. There are significant changes of top-patterns among supporting pro arguments. For instance, arguments mentioning *renewable* energy are supported by arguments about *reliability* but with declining relative support over the years. Further, *climate* takes the spot as the most important aspect in 2019 and 2021 answered with, again, *climate*, and with *renewable energies*. Interestingly, pro-nuclear energy arguments referring to *renewables* are less likely supported *climate*-related replies. A possible explanation is that people supportive of nuclear energy shifted their framing to nuclear energy being necessary to

Table 5. Top 5 support and attack patterns with the most support for each aspect combination, by year. Green arrows indicate a rise in the pattern rank, red arrows indicate a fall, dashes indicate no change in the position.

Pro ← Pro

2017		2019		2021	
Pattern	Support	Pattern	Support	Pattern	Support
renew, renew	8.94%	↑ climate, climate	6.93%	– climate, climate	6.90%
reliability, reliability	8.19%	↓ renew, renew	6.66%	– renew, renew	6.65%
renew, reliability	8.19%	↑ climate, renew	4.69%	↑ costs, costs	5.47%
reliability, renew	6.87%	↓ renew, reliability	4.60%	↓ climate, renew	4.69%
fossil fuels, renew	6.16%	↑ costs, costs	4.54%	↓ renew, reliability	4.43%

Con ← Con

2017		2019		2021	
Pattern	Support	Pattern	Support	Pattern	Support
costs, costs	2.34%	— costs, costs	1.46%	— costs, costs	1.56%
safety, safety	2.14%	↑ renew, renew	1.17%		
renew, renew	1.68%	↓ safety, safety	1.05%		
renew, costs	1.50%				
costs, renew	1.37%				

Con ← Pro

2017		2019		2021	
Pattern	Support	Pattern	Support	Pattern	Support
renew, renew	3.71%	– renew, renew	2.95%	↑ costs, costs	3.23%
costs, renew	3.36%	↑ costs, costs	2.72%	↓ renew, renew	2.49%
renew, reliability	3.03%	↓ costs, renew	2.16%	– costs, renew	2.41%
costs, reliability	2.85%	↑ climate, climate	2.00%	– climate, climate	1.87%
waste, reliability	2.74%	↓ renew, reliability	1.90%	↑ costs, climate	1.75%

Pro ← Con

2017		2019		2021	
Pattern	Support	Pattern	Support	Pattern	Support
renew, renew	2.87%	– renew, renew	2.55%	↑ costs, costs	2.78%
renew, costs	2.36%	↑ climate, costs	2.43%	– climate, costs	2.27%
reliability, reliability	2.23%	↑ costs, costs	2.36%	↓ renew, renew	2.13%
reliability, costs	1.99%	↓ renew, costs	2.24%	– renew, costs	1.93%
reliability, renew	1.97%	↑ climate, climate	1.99%	– climate, climate	1.76%

combat climate change, yet avoided the expression of support for renewables. The pattern *(costs, costs)* steadily climbs up to the top ranks indicating an increasingly important economic framing of the debate in addition to climate aspects.

Support Con ← Con. Chains of con–con arguments are seldom found patterns compared to other combinations. In 2021 the only pattern con-con pattern that has a support of more than 1% is *(costs, costs)*. Similarly to pro–pro patterns, the common patterns tend to reinforce the same aspect.

Attack Con ← Pro. *Reliability* occurs more frequently in 2017 in the top patterns, and only as the pro argument. In 2017, *waste* was part of the most supported patterns, which is the only time for any combination of pro and con stanced tweets. *Costs* is the most occurring con-part of the con-pro patterns, but over the years, it is countered with different pro arguments. While in 2017 *renewables* and *reliability* were used the most for addressing con arguments regarding *costs*, this shifted away slightly from *reliability* to pro arguments regarding *costs.*

Attack Pro ← Con. For Arguments that are predicted with a pro-stance, it can be seen that the overall most common aspect of con-predicted responses is *costs.* Responding with the same aspect is also a prevalent pattern for *renewables*, *reliability*, and *climate. Costs* seems to be an aspect that can be addressed regardless of the pro-aspect that is put forth in favor of nuclear energy. Interestingly, in pro–con chains *costs* only appears in the pro part of the chain starting in 2019 and increases in support to being part of the number one pattern in 2021. This suggests that debates about whether or not nuclear energy is a cost-efficient form of energy production in modern societies intensified significantly.

Longer Patterns. As mentioned earlier, only a small number of patterns longer than two arguments were found in our dataset. Table 6 in the Appendix displays the top five patterns of length $n = 3$ for each year. The table contains exclusively chains of arguments in favor of nuclear energy that mostly reinforce the previously argued aspect. This again suggests that supporters of nuclear energy have a more engaged audience on Twitter compared to opponents of nuclear energy.

4.2 Pattern Mining Vs. Analyzing Distributions

When comparing the aspects of the most common patterns with their distribution, it is evident that the most occurring aspects also occur the most in the top patterns. The important distinction between the two analyses can be seen by analyzing the differences: in 2017, *safety* and *reliability* had near similar occurrence. *Safety* was, however, not discussed in a pro–pro context. Also in 2017, *nuclear waste* was in the top five con–pro patterns, although its proportion among con arguments rose steadily. *Costs* was the most popular aspect addressed by con arguments in 2019 and in 2021, surpassing *safety.* While they had similar proportions in 2019, *safety* is only prevalent in one con–con top-pattern, while pro–con and con-pro chains were more and more overtaken by the discussion revolving around costs. This shows that our method can add a benefit to analyzing social media debates by leveraging the structured information of their conversation trees.

5 Conclusion

In this paper, we have introduced Sequential Pattern Mining on abstract argument representations generated by recent argument mining methods. By mining patterns from a large corpus of German Twitter conversations on nuclear energy, we demonstrated the usefulness for analyzing structured online debates compared to simpler approaches looking at frequencies of isolated events in the data. To construct a transaction dataset for SPM from Twitter conversations, we suggest employing a set of argument mining approaches, in our case argument stance and aspect classification with fine-tuned language models. Combining structural and abstract semantic information in a set of all possible transactions, we found distinctive patterns of argumentation that were not evident from analyzing tweet information in isolation.

5.1 Limitations

While this first application of our method already shows well-interpretable initial results, more validation is indispensable. A first limitation is the validation of prediction results, which we have conducted, but have not evaluated in a structured manner. Since the method relies on the accuracy of the prediction on the dataset, bad classification will falsify the results of the SPM. We have seen cases, where the stance classifier was unable to accurately predict the stance of arguments in their relationship to nuclear energy and also struggling with sarcasm and jokes. However, we expect to receive mostly valid results from pattern mining given the large corpus size. Another problem might stem from the fact that PrefixSpan can find non-contiguous patterns. This might lead to patterns that do not actually indicate attack or support relationships, especially for cases with longer sequences. However, we are quite confident that these issues play a negligible concerning our dataset role given the very large volumes of data that are analyzed using the method and the fact that a majority of the mined conversations contain not more than one reply.

5.2 Future Work

Future work will concentrate on the interpretation and validation of the mined patterns. Since there are many patterns with less support a careful analysis of all attack and support patterns could reveal more insights into the debate. A thorough qualitative analysis is therefore the next step for establishing the method and testing its potential for the computational social science community. This could also be used to verify classification quality and detect potential issues with classification results. While we assume that training a classifier with labeled data still is preferable to using commercial Large-Language-Models such as ChatGPT for highly specific classification tasks, using such LLMs may increase the use of the method for CSS scholars, as extensive labeling and fine-tuning are not necessary. Further research into alternative, potentially better-suited sequence mining algorithms should be conducted, too. Analyzing patterns from constraint-based

SPM approaches that only allow contiguous patterns is an interesting next step for attack and support pattern mining as well as quantifying the chance-corrected statistical significance of the patterns found. Regarding representing and further analyzing attacking and supporting arguments in formalisms dealing explicitly with arguments and argumentation, so-called Abstract Dialectical Frameworks (ADFs) [5] seem to be suitable as they provide sufficient expressive power. Such an approach was suggested in our FAME-project [4] and will be one future research line.

Acknowledgements. This work was funded by the Deutsche Forschungsgemeinschaft (DFG) as part of the project "FAME: A framework for argument mining and evaluation" (project no. 406289255).

Appendix

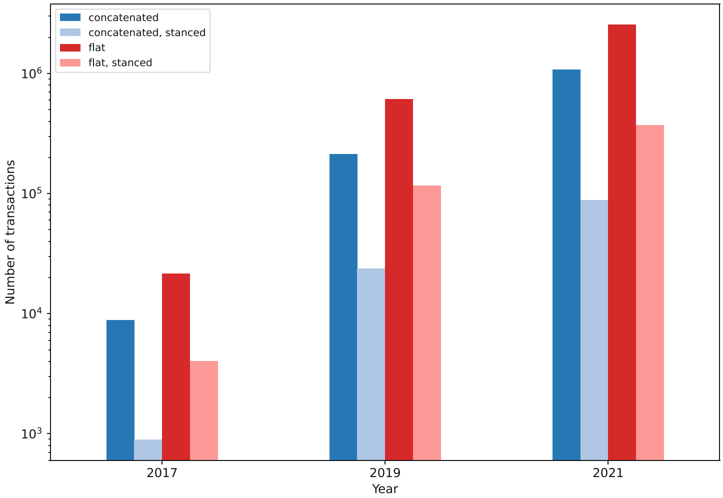


Fig. 5. Number of transactions for *flat* and *concatenated* aspect resolution, in total, and only containing transactions, in which every tweet has either a pro or con stance.

Figure 5 shows the number of transactions for flat and concatenated aspect resolution when including and excluding transactions containing tweets without a predicted stance. The number of transactions is vastly reduced by excluding tweets without a stance. This shows that our method is condensing the dataset significantly, making it more likely that patterns of interest can be found.

Table 6. Top five patterns of length $n = 3$, by year.

Pattern	Support
2017	
(pro, renew), (pro, renew), (pro, reliability)	2.25%
(pro, renew), (pro, reliability), (pro, reliability)	2.21%
(pro, renew), (pro, renew), (pro, renew)	1.83%
(pro, reliability), (pro, reliability), (pro, reliability)	1.81%
(pro, climate), (pro, renew), (pro, reliability)	1.57%
2019	
(pro, renew), (pro, renew), (pro, renew)	1.11%
(pro, climate), (pro, climate), (pro, climate)	0.89%
(pro, renew), (pro, renew), (pro, reliability)	0.77%
(pro, reliability), (pro, renew), (pro, renew)	0.75%
(pro, costs), (pro, costs), (pro, costs)	0.72%
2021	
(pro, renew), (pro, renew), (pro, renew)	1.17%
(pro, climate), (pro, climate), (pro, climate)	1.04%
(pro, costs), (pro, costs), (pro, costs)	0.95%
(pro, renew), (pro, renew), (pro, reliability)	0.74%
(pro, fossil fuels), (pro, renew), (pro, renew)	0.73%

References

1. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Proceedings of ICDE 1995, pp. 3–14 (1995)
2. Ajjour, Y., Alshomary, M., Wachsmuth, H., Stein, B.: Modeling frames in argumentation. In: Proceedings of EMNLP-IJCNLP 2019, pp. 2922–2932. ACL, Hong Kong, China (2019)
3. Ajjour, Y., Kiesel, J., Stein, B., Potthast, M.: Topic ontologies for arguments. In: Findings of the Association for Computational Linguistics: EACL 2023, pp. 1411–1427. ACL, Dubrovnik, Croatia (2023)
4. Baumann, R., Wiedemann, G., Heinrich, M., Hakimi, A.D., Heyer, G.: The road map to FAME: a framework for mining and formal evaluation of arguments. Datenbank-Spektrum **20**(2), 107–113 (2020)
5. Brewka, G., Woltran, S.: Abstract dialectical frameworks. In: Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR, Toronto, Ontario, Canada (2010)
6. Chakrabarty, T., Hidey, C., Muresan, S., McKeown, K., Hwang, A.: AMPERSAND: argument Mining for PERSuAsive oNline Discussions. In: Proceedings of EMNLP-IJCNLP 2019, pp. 2933–2943. ACL, Hong Kong, China (2019)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceeding of NAACL 2019, pp. 4171–4186. ACL, Minneapolis, Minnesota (2019)

8. Friedman, R., Dankin, L., Hou, Y., Aharonov, R., Katz, Y., Slonim, N.: Overview of the 2021 key point analysis shared task. In: Proceedings of ArgMining 2021, pp. 154–164. ACL, Punta Cana, Dominican Republic (2021)
9. Heinisch, P., Cimiano, P.: A multi-task approach to argument frame classification at variable granularity levels. IT - Inf. Technol. **63**(1), 59–72 (2021)
10. Lapesa, G., Blessing, A., Blokker, N., Dayanik, E., Haunss, S., Kuhn, J.: Analysis of political debates through newspaper reports: methods and outcomes. Datenbank-Spektrum **20**(2), 143–153 (2020)
11. Lapesa, G., Blessing, A., Blokker, N., Dayanik, E., Haunss, S., Kuhn, J.: DEbateNet-mig15: tracing the 2015 immigration debate in Germany over time. In: Proceedings of LREC 2020, pp. 919–927. ELRA, Marseille, France (2020)
12. Lawrence, J., Reed, C.: Argument mining: a survey. Comput. Linguist. **45**(4), 765–818 (2019)
13. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach. CoRR abs/1907.11692 (2019)
14. Mirzakhmedova, N., Kiesel, J., Al-Khatib, K., Stein, B.: Unveiling the power of argument arrangement in online persuasive discussions. In: Findings of the Association for Computational Linguistics: EMNLP 2023. ACL (2023)
15. Nguyen, H., Litman, D.: Context-aware argumentative relation mining. In: Proceedings of ACL 2016, pp. 1127–1137. ACL, Berlin, Germany (2016)
16. Pei, J., et al.: PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth. In: Proceedings of IDCE 2001, pp. 215–224 (2001)
17. Rieger, J., Yanchenko, K., Ruckdeschel, M., von Nordheim, G., Kleinen-von Königslöw, K., Wiedemann, G.: Few-shot learning for automated content analysis: efficient coding of arguments and claims in the debate on arms deliveries to Ukraine. Stud. Commun. Media **13**(1), 72–100 (2024)
18. Ruckdeschel, M., Wiedemann, G.: Boundary detection and categorization of argument aspects via supervised learning. In: Proceedings of ArgMining 2022, pp. 126–136. COLING, Online and in Gyeongju, Republic of Korea (2022)
19. Ruckdeschel, M., Wiedemann, G.: Argument aspect corpus (2023). https://doi.org/10.5281/zenodo.7525183
20. Scheffler, T., Aktaş, B., Das, D., Stede, M.: Annotating shallow discourse relations in Twitter conversations. In: Proceedings of W19-27 2019, pp. 50–55. ACL, Minneapolis, MN (2019)
21. Song, N., Cheng, H., Zhou, H., Wang, X.: Argument structure mining in scientific articles: a comparative analysis. In: 2019 ACM/IEEE Joint Conference on Digital Libraries (JCDL), pp. 339–340 (2019)
22. Stab, C., Gurevych, I.: Parsing argumentation structures in persuasive essays. Comput. Linguist. **43**(3), 619–659 (2017)
23. Stab, C., Miller, T., Rai, P., Schiller, B., Gurevych, I.: UKP sentential argument mining corpus (2018). https://doi.org/10.18653/v1/D18-1402
24. Trautmann, D.: Aspect-based argument mining. In: Proceedings of ArgMining 2020, pp. 41–52. ACL, Online (2020)
25. Vecchi, E.M., Falk, N., Jundi, I., Lapesa, G.: Towards argument mining for social good: a survey. In: Proceedings of ACL-IJCNLP 2021, pp. 1338–1352. ACL, Online (2021)
26. Wang, K., Chua, T.S.: Exploiting salient patterns for question detection and question retrieval in community-based question answering. In: Proceedings of COLING 2010. COLING 2010, pp. 1155–1163. Org. Committee, Beijing, China (2010)

27. Yamamoto, K., Kudo, T., Tsuboi, Y., Matsumoto, Y.: Learning sequence-to-sequence correspondences from parallel corpora via sequential pattern mining. In: Proceedings of W03-0300 2003, pp. 73–80 (2003)
28. Yu, X., Blanco, E., Hong, L.: Hate speech and counter speech detection: conversational context does matter. In: Proceedings of NAACL 2022, pp. 5918–5930. ACL, Seattle, United States (2022)

Cluster-Specific Rule Mining for Argumentation-Based Classification

Jonas Klein(✉), Isabelle Kuhlmann, and Matthias Thimm

FernUniversität in Hagen, Hagen, Germany
{jonas.klein,isabelle.kuhlmann,matthias.thimm}@fernuni-hagen.de

Abstract. We present a multi-step classification approach that combines classical machine learning methods with computational models for argumentation. In the first step, the dataset is divided into different groups using a clustering algorithm. In the second step, we employ rule-learning algorithms to extract frequent patterns and rules from each resulting cluster. In the last step, we interpret the rules as the input for structured argumentation approaches. Given a new observation, we first assign it to one of the previously generated clusters. Subsequently, the classification of the observation is determined by formulating arguments based on the respective cluster-specific rules for the different classes. Finally, the justification status of the arguments is determined using the argumentative inference method of the structured argumentation approach.

Keywords: Argumentation · Classification · Rule Mining

1 Introduction

Classification is a widely known problem in the field of artificial intelligence. In recent years, machine learning approaches, in particular different forms of neural networks, have made substantial progress in solving classification tasks for a diverse range of domains—such as computer vision [15], text processing [10], or graph theory [19]. However, although current machine learning methods for classification purposes may yield remarkably accurate results, they are still not *guaranteed* to be correct, and they are not inherently explainable, i. e., no form of justification or rationale is provided. On the other hand, the need for explainable methods is becoming increasingly relevant [4].

To address the problem of lacking explainability in machine learning-based classification approaches, Thimm and Kersting [16] propose an approach that combines machine learning with computational models of argumentation [5]. To be precise, the authors suggest a two-step procedure: first, a rule learning algorithm is applied to extract rules from a given dataset; in the second step, the learned rules are used as input for a structured argumentation system, which then yields a justification status for each class, given a new observation. Thus, this approach does not only deliver classifications but also explanations thereof.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 57–67, 2024.
https://doi.org/10.1007/978-3-031-63536-6_4

Further, expert knowledge (in the form of additional arguments) can easily be incorporated into the reasoning process.

Thimm and Kersting [16] presented some preliminary experimental results in their study: on the *Animals with Attributes* (AwA) dataset, about 30 % of the instances were classified correctly, while the remaining 70 % were deemed "undecided".[1] In this paper, we build explicitly on these results and present an extended approach that likewise includes a rule mining step and an argumentation-based classification step, which introduces a clustering technique for more targeted rule mining. More specifically, the clustering step reduces the number of mined rules to make them more purposeful and additionally counteracts the extraction of contradictory rules. In an experimental analysis, we show that our method can achieve a significantly higher accuracy of 71 % on the AwA dataset. To corroborate our observations, we consider additional datasets. Furthermore, we demonstrate that the procedure introduced in this work is potentially significantly more resource-efficient than the approach proposed by Thimm and Kersting.

2 Background

The three main ingredients of the approach presented in this paper are (1) a clustering algorithm, (2) a rule mining algorithm, and (3) a structured argumentation method. Although the choice of each component is generally flexible, we select some concrete instantiations of each component as an example. For the clustering part, we use a simple *k-modes* algorithm [12], which is aimed at clustering categorical variables. As the rule mining algorithm we use *association rule mining*, and as the structured argumentation approach, following [16], we use *defeasible logic programming* [9]. Both latter formalisms are outlined below.

Association Rule Mining. Data mining generally encompasses methods for extracting non-trivial patterns from a given dataset. *Association rule mining* [3] aims to uncover interesting relationships among items within extensive databases. Consider $I = \{I_1, I_2, \ldots, I_m\}$ as a set comprising m distinct attributes. Let T be a transaction containing a set of items such that $T \subseteq I$, and let D be a database with various transaction records T_s. An *association rule* takes the form of $X \Rightarrow Y$, where $X, Y \subseteq I$ represent sets of items known as *itemsets*, and $X \cap Y = \emptyset$. In this context, X is referred to as the *antecedent*, and Y is termed the *consequent*. The rule $X \Rightarrow Y$ signifies that the presence of X implies the presence of Y. Association rules rely on two fundamental criteria of interestingness: *support* and *confidence*. These criteria help identify relationships and rules by revealing frequently occurring if/then patterns. To be considered, association rules typically must meet both a user-specified minimum *support* and

[1] Note that the authors used defeasible logic programming (DeLP) [9] as the structured argumentation approach, and DeLP does not only use "yes" and "no" as answers, but also "undecided".

a user-specified minimum *confidence* simultaneously. The *support* of an association rule is defined as the fraction of records that contain $X \cup Y$ relative to the total number of records in the database. The *confidence* of an association rule is defined as the fraction of the number of transactions that contain $X \cup Y$ relative to the total number of records that contain X. For the approach presented here, we use *FP-Growth* [11] as a rule miner.

Defeasible Logic Programming. The core idea behind *defeasible logic programming* (DeLP) [9] is to combine concepts from logic programming and defeasible argumentation to allow for dealing with incomplete or contradictory data. A defeasible logic program (de.l.p.) consists of facts and rules which are divided into *strict* rules of the form $l \leftarrow B$ and *defeasible* rules of the form $l \prec B$, with l being a single literal and B a set of literals. Moreover, a *fact* is a single literal (i. e., an atom a or a negated atom $\neg a$). Thus, formally, a de.l.p. $\mathcal{P} = (\mathit{\Pi}, \mathit{\Delta})$ consists of a set $\mathit{\Pi}$ of facts and strict rules, and a set $\mathit{\Delta}$ of defeasible rules. Further, a literal l is *derivable* by some set of rules R (i. e., $R \mid\!\sim l$) if it is derivable following the classical rule-based understanding. If both $R \mid\!\sim l$ and $R \mid\!\sim \neg l$, then R is *contradictory*. Conventionally, $\mathit{\Pi}$ is non-contradictory. Further, if $R \mid\!\sim l$, and $R \not\mid\!\sim \bot$, we call the literal l *consistently derivable* (denoted as $R \mid\!\sim^c l$).

For a de.l.p $\mathcal{P} = (\mathit{\Pi}, \mathit{\Delta})$ and a literal l, a tuple $\langle \mathcal{A}, l \rangle$ (with $\mathcal{A} \subseteq \mathit{\Delta}$) is an *argument* for l iff $\mathit{\Pi} \cup \mathcal{A} \mid\!\sim^c l$ and $\mathcal{A}$ is minimal wrt. set inclusion. Further, $\langle \mathcal{B}, q \rangle$ is a *subargument* of $\langle \mathcal{A}, l \rangle$ iff $\mathcal{B} \subseteq \mathcal{A}$. We refer to $\langle \mathcal{A}_1, l_1 \rangle$ as a *counterargument* to $\langle \mathcal{A}_2, l_2 \rangle$ at literal l, iff there is a subargument $\langle \mathcal{A}, l \rangle$ of $\langle \mathcal{A}_2, l_2 \rangle$ with $\mathit{\Pi} \cup \{l, l_1\}$ being contradictory. To deal with counterarguments, we use the *generalized specificity* relation $\succ$ as a formal comparison criterion among arguments. According to this criterion, an argument is preferred over another, if (1) it has a greater information content and is thus more precise, or (2) it uses fewer rules and is thus more concise (see Garcia and Simari [9] for a formal definition and further discussion). We call $\langle \mathcal{A}_1, l_1 \rangle$ a *defeater* of $\langle \mathcal{A}_2, l_2 \rangle$ iff there is a subargument $\langle \mathcal{A}, l \rangle$ of $\langle \mathcal{A}_2, l_2 \rangle$ such that $\langle \mathcal{A}_1, l_1 \rangle$ is a counterargument of $\langle \mathcal{A}_2, l_2 \rangle$ at literal l and either $\langle \mathcal{A}_1, l_1 \rangle \succ \langle \mathcal{A}, l \rangle$ (*proper defeat*) or $\langle \mathcal{A}_1, l_1 \rangle \not\succ \langle \mathcal{A}, l \rangle$ and $\langle \mathcal{A}, l \rangle \not\succ \langle \mathcal{A}_1, l_1 \rangle$ (*blocking defeat*).

A finite sequence of arguments $\mathit{\Lambda} = [\langle \mathcal{A}_1, l_1 \rangle, \ldots, \langle \mathcal{A}_m, l_m \rangle]$ is an *acceptable argumentation line* iff (1) every $\langle \mathcal{A}_i, l_i \rangle$ with $i > 1$ is a defeater of $\langle \mathcal{A}_{i-1}, l_{i-1} \rangle$ and if $\langle \mathcal{A}_i, l_i \rangle$ is a blocking defeater of $\langle \mathcal{A}_{i-1}, l_{i-1} \rangle$ and $\langle \mathcal{A}_{i+1}, l_{i+1} \rangle$ exists, then $\langle \mathcal{A}_{i+1}, h_{i+1} \rangle$ is a proper defeater of $\langle \mathcal{A}_i, h_i \rangle$, (2) the sets $\mathit{\Pi} \cup \mathcal{A}_1 \cup \mathcal{A}_3 \cup \ldots$ and $\mathit{\Pi} \cup \mathcal{A}_2 \cup \mathcal{A}_4 \cup \ldots$ are non-contradictory, and (3) there exists no $\langle \mathcal{A}_k, l_k \rangle$ as a subargument of $\langle \mathcal{A}_i, l_i \rangle$ with $i < k$. Thus, intuitively, an argumentation line forms a sequence of arguments, in which each $\langle \mathcal{A}_i, l_i \rangle$ defeats its predecessor $\langle \mathcal{A}_{i-1}, l_{i-1} \rangle$. Moreover, since an argument $\langle \mathcal{A}_i, l_i \rangle$ defeats $\langle \mathcal{A}_{i-1}, l_{i-1} \rangle$, and therefore reinstates $\langle \mathcal{A}_{i-2}, l_{i-2} \rangle$, the sets $\mathit{\Pi} \cup \mathcal{A}_1 \cup \mathcal{A}_3 \cup \ldots$ and $\mathit{\Pi} \cup \mathcal{A}_2 \cup \mathcal{A}_4 \cup \ldots$ must be non-contradictory in order for the argumentation line to be acceptable. To avoid circular argumentation, we also need to ensure that no subarguments are reintroduced in the same argumentation line.

Finally, a literal l is *warranted* if there is an argument $\langle \mathcal{A}, l \rangle$ which is non-defeated in the end. To decide whether $\langle \mathcal{A}, l \rangle$ is defeated or not, every acceptable argumentation line starting with $\langle \mathcal{A}, l \rangle$ has to be considered. The answer is to a DeLP query is Yes if l is warranted, and No if $\neg l$ is warranted. Otherwise, the answer is Undecided.

3 Cluster-Specific Rule Mining

The approach proposed in this work is an extension of the *argumentation-based classification* approach (AbC) described by Thimm and Kersting [16]. The AbC approach consists of two steps: (1) Mining of association rules from a given dateset and (2) performing classification using the generated rules as an input to a structured argumentation approach. During the initial phase, algorithms for rule mining are employed to identify frequent patterns and rules from a specified dataset. The result of this step yields a substantial number of rules [17]. However, these rules cannot be directly applied to classification since they often exhibit inconsistencies. Hence, in the subsequent phase, these rules are used as input to structured argumentation methods, such as DeLP. Employing the argumentative inference procedures inherent in these approaches, the classification of the new observation is executed by formulating arguments based on these rules and evaluating their justification status. Using argumentation techniques enables the creation of classifiers explicitly designed to explain their decisions, thus meeting the contemporary demand for explainable AI. These classifiers are able to explain the reasons for favoring arguments supporting the conclusion over counterarguments.

We extend the original two-step argumentation-based classification approach AbC to a multi-step classification method, that combines traditional machine learning methods with structured argumentation. To be precise, we introduce two additional steps. Firstly, we perform a clustering of the input data, resulting in groups of instances with similar properties. Secondly, a feature selection is carried out for each cluster to identify the most informative features for the prediction of the target variable. Subsequently, these features are used to generate cluster-specific association rules for each cluster. Since the number of generated rules significantly influences the classification time, this approach leads to significantly shorter runtimes and is more resource-efficient. In addition, grouping instances with similar properties leads to discovering relationships that are difficult to detect when looking at the entire dataset. This improves the capability to classify datasets where a naive approach may not extract enough rules. Moreover, the generated rule set is more consistent due to the similarity of the instances within a cluster and the emphasis on meaningful features, improving the decidability of instances and thus reducing the number of undecidable instances. In general, the presented approach consists of four steps: (1) Clustering the input data, (2) cluster-specific feature importance analysis to select the most informative features, (3) cluster-specific association rule mining based on these features, and (4) classification of new observations by assigning them to a cluster and using the cluster-specific rules. Each step is outlined in more detail below.

Clustering. First, the input data is divided into k groups based on all features (including the target feature), using the *k-modes* [12] algorithm. The *k-modes* clustering algorithm modifies the well-known *k-means* clustering method for partitioning a dataset into distinct groups or clusters based on categorical data. This step aims to divide the input data into smaller, more manageable groups with similar properties to reduce the running time of the rule mining algorithm, reduce the number of rules generated, improve the detection of otherwise hard to find relations and improve the quality of the rules.

Feature Selection. This step conducts a feature importance analysis to find the most informative features for classifying the target variable within a cluster using the mutual information score. Mutual information quantifies the relationship between two random variables with a value that is always non-negative, indicating their dependency level. This value is zero exclusively when the two variables are independent, with larger values indicating a greater dependency. The score calculation is based on entropy estimation using distances from k-nearest neighbors, as outlined in [13,14]. After calculating the scores for each feature, the top k features are selected. The selected cluster-specific features are used as the input for the rule miner in the next step. The association rule mining step is massively accelerated by reducing the number of features and discarding features with little expressiveness. Furthermore, only the most relevant features are used for rule mining, leading to fewer, more meaningful rules.

Association Rule Mining. This step generates cluster-specific association rules for each previously generated cluster based on the most important selected features. In this work, we use the *FP-Growth* [11] algorithm. In principle, however, any rule mining algorithm is usable. To generate rules from the truth values of the features of an instance, these are represented as a set of ground literals. For example, for a dataset of animals with the attributes *swims*, *black*, and *arctic*, the attributes of a dolphin would be represented as $\mathit{swims}(\mathit{dolphin})$, $\neg \mathit{black}(\mathit{dolphin})$, and $\neg \mathit{arctic}(\mathit{dolphin})$. The output of the rule mining algorithm is a set of association rules such as $\mathit{flippers}(X) \rightarrow \mathit{ocean}(X)$, which can be interpreted as "animals with flippers live in the ocean". Subsequently, the created rules are filtered according to the method of Thimm and Kersting [16]: Rules with more than one element in the conclusion and more than three elements in the body are discarded. All rules with confidence value 1 are interpreted as strict; the remaining rules are interpreted as defeasible.[2] Occasionally, no or not enough cluster-specific rules are generated for the target variable, resulting in instances assigned to this cluster not being able to be classified. To prevent this, we implemented an adaptive rule mining process, which iteratively adjusts the confidence and support values until at least one rule for the target variable is generated.

[2] We followed this procedure to ensure the best comparability with the original approach. A systematic analysis of different rule filtering techniques and strict/defeasible thresholds is out of the scope of this work and saved for future work.

Classification. In the classification step, the cluster-specific rules are used as input to the structured argumentation approach DeLP (see Sect. 2). To classify a new instance, it is first assigned to a cluster using the previously trained clustering algorithm to determine the classification rule set. Since the value of the target variable is unknown, the assignment is performed twice, whereby (1) the target variable is assumed to be positive and (2) it is assumed to be negative. If, for example, one aims to classify the edibility of a mushroom with the classes *edible* and *poisonous*, an unseen mushroom is once assumed to be *edible* and another time assumed to be *poisonous*. Two cases can occur: The mushroom is either assigned to the same cluster in both cases or to different clusters. In the first case, the rules of the corresponding cluster are applied, and the classification is carried out. In the second case, two classifications are performed with the different rules of the respective cluster. Since two different sets of rules from different clusters are used, and different assumptions are made about the class of the target variable, conflicting classifications may occur. For example, a mushroom m is assigned to cluster C_0 for the negative assumption (*poisonous*) and to cluster C_1 for the positive assumption (*edible*). The query *poisonous*(m) returns the answer UNDECIDED for C_0. For C_1, the answer is YES. Since the results do not match, one of the two answers must be selected. In general, two types of conflicts can arise: (1) The rules of one cluster return UNDECIDED, and the rules of the other cluster return a concrete answer YES/NO, or (2) one cluster returns YES and the other NO. The first conflict is resolved by choosing the concrete answer (YES/NO) as the final result. In the second case, the answer of the rule set with the higher average confidence is used. If the average confidence matches, the average support is used as a tiebreaker.

4 Experimental Analysis

In this section, we present the results of an experimental analysis, in which we compare our approach[3] to AbC [16] in terms of the classification performance on five different datasets. Below, we describe the experimental setup and subsequently discuss our findings.

Datasets and Setup. We use five well-known categorical datasets for binary classification as training and test data: *Animals with Attributes*, *Zoo*, *Mushrooms*, *Car Evaluation*, and *Congressional Voting Records*.

All categorical features that are not already in binary form were one-hot encoded by converting each feature into as many 0/1 features as there are different values. For a dataset with, for example, the feature *safety*, which has two different values, low and high, two new dummy features, *safety_low* and *safety_high*, have been introduced. For each instance, the feature's value was then replaced by the corresponding one-hot encoding. Records with missing values were excluded.

We make use of the following five datasets.

[3] https://github.com/jklein94/Cluster-Specific-Rule-Mining-for-Argumentation-Based-Classification.

Animals with Attributes. The *Animals with Attributes* (*AwA*) dataset consists of 50 different animals with 85 boolean-valued attributes. The dataset was randomly split into 90% training data and 10% test data. The *Zoo* [8] dataset is similar to the *AwA* dataset. It contains 101 instances of animals with 16 attributes. The dataset was randomly split into 80% training data and 20% test data.

Mushrooms. The *Mushrooms* [2] dataset comprises descriptions of imaginary samples representing 23 species of mushrooms. Each species is categorized as either edible, poisonous, or of uncertain edibility. The latter category was merged with the poisonous one. The dataset initially consists of 8124 instances with 22 categorical features. After the data cleaning and feature encoding, the dataset contains 5644 instances with 99 features and was randomly split into 90% training data and 10% test data.

Car Evaluation. The *Car Evaluation* (*Car*) [6] dataset was derived from a simple hierarchical decision model. The original dataset contains 1728 instances with 6 categorical features. After one-hot encoding, a total of 22 features resulted. No instance was excluded. The classification target is determining whether a car exhibits a low safety standard. The dataset was randomly split into 80% training data and 20% test data.

Congressional Voting Records. The *Congressional Voting Records* (*Congress*) dataset [1] consists of 1984 US Congressional Voting Records for each of the U.S. House of Representatives Congressmen. Initially, it contains 435 instances and 16 features. After removing the records with missing values and encoding the features, 232 instances with 33 features remain. The classification target is to determine which party (Democratic or Republican) a congressman voted for. The dataset was randomly split into 80% training and 20% test data.

We repeated the classification five times for each dataset according to the procedure described in Sect. 3. The number of randomly initialized clusters was set to seven. For each cluster, the top four features were selected. We set the minimum support of the rule mining algorithms to 0.7 and the minimum confidence to 0.9.[4] We use the accuracy, percentage of undecided instances, and percentage of decided, but falsely classified instances to evaluate the performance of the proposed approach. The mean result of the five runs is reported. Note that we randomly selected ten attributes for the *AwA* dataset as target variables. The average of the metrics across all ten selected attributes is reported. We randomly selected three target variables for the *Zoo* dataset. The results for each selected attribute are reported individually.

Results. The results in Table 1 show that AbC could not classify even one instance for five of the seven scenarios. To be precise, for the classification of

[4] The values used showed promising results in preliminary experiments, achieving a good balance between the number of generated rules, classification performance, and runtime. A systematic analysis of the parameters is beyond the scope of this work and will be part of future work.

Table 1. Overview of accuracy (ACC), undecidable instances (UNDEC), and decided but falsely classified instances for our approach and the AbC approach. The results of AbC for the *AwA* dataset are those reported in [16].

Name	Ours			AbC		
	ACC	UNDEC (%)	False (%)	ACC	UNDEC (%)	False (%)
AwA	0.71	17.20	12.0	0.30	70.0	0.00
Zoo_eggs	1.00	0.00	0.00	0.00	100.0	0.00
Zoo_milk	0.96	0.95	2.86	0.00	100.0	0.00
Zoo_fins	0.92	2.86	4.76	0.86	9.52	4.76
Mushrooms	0.88	10.80	1.13	-	-	-
Car	0.82	15.26	2.77	0.0	100.0	0.00
Congress	0.87	4.68	8.09	0.00	100.0	0.00

Zoo_eggs, *Zoo_milk*, *Car*, and *Congress* all test instances were answered as Undecided, leading to an accuarcy of 0. Our approach, on the other hand, consistently achieves high accuracies ranging from 0.82 (*Cars*) to 1.0 (*Zoo_eggs*). For *Zoo_fins*, both approaches show 4.76 % of falsely classified instances. However, our approach exhibits a significantly lower proportion of Undecided instances, reflected in a higher overall accuracy of 0.92 compared to AbC (0.86). In addition, in our experiments, AbC created a very large number of rules for the *AwA* dataset, which precluded classification in a reasonable time, which is why we rely on the results reported in [16]. Although the results can only be compared to a limited extent, our cluster-specific approach shows significantly higher accuracy (0.71 vs. 0.3) and a significantly smaller proportion of undecidable instances (17.2 % to 70 %) than AbC. The most extensive dataset *Mushrooms* could not be classified by AbC because it ran out of memory in the rule mining step. Our method achieves an accuracy of 0.88, with 10.8 % of instances remaining undecided and a low 1.13 % false classification rate.

5 Limitations

The approach presented in this paper achieves promising results in terms of accuracy and the reduction of undecidable instances. However, there is still room for improvement. In the following, we will discuss some main limitations of the proposed approach.

Rule Generation Control. The classification performance heavily depends on the generated rules. However, direct control over the rule-generation process is limited to setting support and confidence thresholds. Another way to influence rule generation is through clustering and feature selection. Finding the best parameters for the clustering and feature selection steps is a non-trivial task that ultimately comes down to trial and error as it is very dataset-dependent.

Computational Overhead. Compared to traditional machine learning methods, our approach can result in longer classification times due to its multi-step nature. Each step has a computational overhead, and the runtime of the clustering algorithm, the rule mining algorithm, and the DeLP implementations significantly influence our approach's runtime performance.

Classification Tasks and Data Types. Our method's design primarily targets binary classification tasks, focusing on handling categorical variables. In its current configuration, achieving multi-class classification necessitates multiple invocations of the classification pipeline—one for each class. This requirement can significantly heighten computational demands, potentially detracting from overall performance efficiency. Moreover, the approach's specialization in categorical variables necessitates that numeric features undergo a binning process to be transformed into categorical equivalents. This transformation can lead to an exponential increase in the number of features, substantially expanding the feature space.

6 Conclusion

In this work, we presented a new approach to argumentation-based classification. Building on the preliminary results of Thimm and Kersting [16], we developed a multi-step classification approach that combines classical machine learning methods with approaches to (structured) argumentation. In an experimental analysis, we examined the classification performance on five different dataset and showed that our cluster-specific rule mining approach achieves significantly better accuracies and lower numbers of undecidable instances than the original AbC approach. In future work, we aim to explore the influence of different configurations for the clustering, feature selection, and rule mining steps and their impact on classification performance. Furthermore, broadening the scope of evaluation to encompass datasets of increased complexity and diversity and a comparative analysis with other argumentation-based methods like ABALearn [18] and AA-CBR [7], other symbolic learners and traditional machine learning approaches would be of great interest. Moreover, efforts to improve scalability and computational efficiency are paramount. Optimizing the approach to handle larger datasets efficiently without sacrificing explainability or classification accuracy is critical for practical use. Finally, extending our methodology to efficiently tackle multi-class classification tasks and accommodate diverse data types, including continuous and multi-modal datasets, represents a significant frontier for exploration.

Acknowledgement. The research reported here was partially supported by the Deutsche Forschungsgemeinschaft (grant 375588274).

References

1. Congressional Voting Records. UCI Machine Learning Repository (1987). https://doi.org/10.24432/C5C01P
2. Mushroom. UCI Machine Learning Repository (1987). https://doi.org/10.24432/C5959T
3. Agrawal, R., Imieliński, T., Swami, A.: Mining association rules between sets of items in large databases. In: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207–216 (1993)
4. Angelov, P.P., Soares, E.A., Jiang, R., Arnold, N.I., Atkinson, P.M.: Explainable artificial intelligence: an analytical review. Wiley Interdiscip. Rev. Data Mining Knowl. Disc. **11**(5), e1424 (2021)
5. Baroni, P., Caminada, M., Giacomin, M.: An introduction to argumentation semantics. Knowl. Eng. Rev. **26**(4), 365–410 (2011)
6. Bohanec, M.: Car Evaluation. UCI Machine Learning Repository (1997). https://doi.org/10.24432/C5JP48
7. Cocarascu, O., Stylianou, A., Čyras, K., Toni, F.: Data-empowered argumentation for dialectically explainable predictions. In: ECAI 2020, pp. 2449–2456. IOS Press (2020)
8. Forsyth, R.: Zoo. UCI Machine Learning Repository (1990). https://doi.org/10.24432/C5R59V
9. García, A.J., Simari, G.R.: Defeasible logic programming: an argumentative approach. Theory Pract. Logic Program. **4**(1–2), 95–138 (2004)
10. Gasparetto, A., Marcuzzo, M., Zangari, A., Albarelli, A.: A survey on text classification algorithms: from text to predictions. Information **13**(2), 83 (2022)
11. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. ACM SIGMOD Rec. **29**(2), 1–12 (2000)
12. Huang, Z.: Extensions to the k-means algorithm for clustering large data sets with categorical values. Data Min. Knowl. Disc. **2**(3), 283–304 (1998)
13. Kraskov, A., Stögbauer, H., Grassberger, P.: Estimating mutual information. Phys. Rev. E **69**(6), 066138 (2004)
14. Ross, B.C.: Mutual information between discrete and continuous data sets. PLoS ONE **9**(2), e87357 (2014)
15. Sharma, S., Guleria, K.: Deep learning models for image classification: comparison and applications. In: 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering, pp. 1733–1738. IEEE (2022)
16. Thimm, M., Kersting, K.: Towards argumentation-based classification. In: Logical Foundations of Uncertainty and Machine Learning, Workshop at IJCAI (2017)
17. Thimm, M., Rienstra, T.: Approximate reasoning with ASPIC+ by argument sampling. In: Proceedings of the Third International Workshop on Systems and Algorithms for Formal Argumentation (SAFA 2020), September 2020
18. Tirsi, C., Proietti, M., Toni, F.: ABALearn: an automated logic-based learning system for ABA frameworks. In: Basili, R., Lembo, D., Limongelli, C., Orlandini, A. (eds.) AIxIA 2023 – Advances in Artificial Intelligence. AIxIA 2023. LNCS, vol. 14318, pp. 3–16. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-47546-7_1
19. Xiao, S., Wang, S., Dai, Y., Guo, W.: Graph neural networks in node classification: survey and evaluation. Mach. Vis. Appl. **33**, 1–19 (2022)

Debate Analysis and Deliberation

Automatic Analysis of Political Debates and Manifestos: Successes and Challenges

Tanise Ceron[1](✉), Ana Barić[2], André Blessing[1], Sebastian Haunss[3], Jonas Kuhn[1], Gabriella Lapesa[4,5], Sebastian Padó[1], Sean Papay[1], and Patricia F. Zauchner[3]

[1] IMS, University of Stuttgart, Stuttgart, Germany
tanise.ceron@ims.uni-stuttgart.de
[2] FER, University of Zagreb, Zagreb, Croatia
[3] SOCIUM, University of Bremen, Bremen, Germany
[4] GESIS Cologne, Cologne, Germany
[5] DIID, HHU Düsseldorf, Düsseldorf, Germany

Abstract. The opinions of political actors (e.g., politicians, parties, organizations) expressed through *claims* are the core elements of political debates and decision-making. Political actors communicate through different channels: parties publish manifestos for major elections, while individual actors make statements on a day-to-day basis as reflected in the media. These two channels offer different approaches for analysis: Manifestos, on the one hand, are useful to characterize the parties' positions at a global ideological level over time. In contrast, individual statements can be collected to analyze debates in particular policy domains on a fine-grained level, in terms of individual actors and claims. In this article, we summarize a series of studies we have carried out. We apply NLP-driven (semi-)automatic analyses on these two channels and compare their potentials and challenges. The fine-grained analysis yields rich insights into the communication but comes at the cost of three challenges: (a) a substantial hunger for manual annotation, introducing practical hurdles for analysis both within and across languages; (b) difficulties in claim classification arising from the uneven frequency distribution over the theory-based annotation schemas; (c) the need to map actor mentions onto canonical versions. Manifesto-based analysis avoids these challenges to a substantial extent when a more coarse-grained analysis of party positions is sufficient. We highlight the benefits and challenges of both approaches, and conclude by outlining perspectives for addressing the challenges in future research.

Keywords: Claim identification · discourse network analysis · party positioning · argument mining

1 Introduction

Political decision-making in democracies is generally preceded by political debates taking place in parliamentary forums (committees, plenary debates),

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 71–88, 2024.
https://doi.org/10.1007/978-3-031-63536-6_5

different public spheres (e.g., newspapers, television, social media), and in the exposition of political ideologies in party manifestos [44,46]. In these debates, various actors voice their positions and beliefs, make claims and try to advance their agendas. Political scientists have therefore developed a range of methods to analyze these debates in the dual goal of understanding democratic decision-making and identifying influential actors and important arguments driving the development of these debates. Two prominent ones are as follows:

(a.) To obtain a maximally informative picture, we can identify the claims and actors involved in a given debate, combining political claims analysis [23] and network science, and represent them as *discourse networks* [26,27]. This permits researchers to capture structural aspects of political debates, investigating and reconstructing debates in a fine-grained manner and understanding the reasons why some claims prevail and others fail.

(b.) The more traditional approach in the political science tradition is to abstract away from the details of a given debate and assess positions and beliefs of political actors at the aggregate level of party positions, namely analyzing manifestos. This provides much less detail but focuses on the arguably most important group of political actors and their respective ideologies. Shifts in ideology allow understanding the change of opinions within a party and their electorate [3]. This approach also allows for direct access to actors' opinions as in comparison with news that goes through a selection of actors and decisions when reported in the media outlets.

In this article, we present an overview of the main contributions from a series of studies that aimed at assessing whether these two approaches can be conducted more efficiently using methods from natural language processing (NLP). We start in Sect. 2 with the more complex approach (a), conceptualizing discursive exchanges as discourse networks. Our goal here is to assess how NLP can help to overcome the roadblocks that studies in this perspective are facing because of the time- and labor-intensive annotation required by detailed analyses of political discourse. Then, in Sect. 3, we switch perspective to approach (b), adopting instead the goal of characterizing party positions at the global, ideological level. We demonstrate that this task does not require a full-fledged discourse network analysis, can do with very coarse-grained content categories, and that hardly any manual annotation is necessary. We highlight the benefits and challenges of both approaches, and conclude by outlining perspectives for addressing the challenges in future research.

2 Fine-Grained Analysis of Political Discourse

Our starting point for the first approach is political debates as they are represented in newspaper articles. In these articles, journalists report on claims and positions of all kinds of actors participating in public debates. We conceptualize these discursive interactions as discourse networks [26] — (dynamic) bipartite

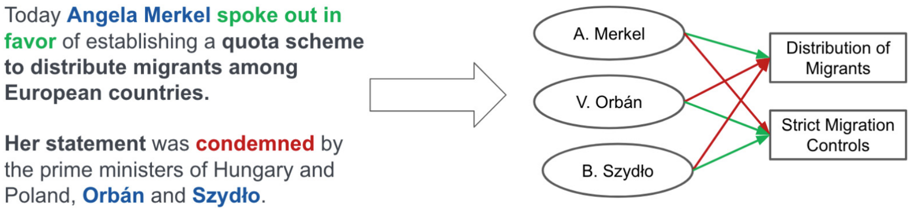

Today Angela Merkel spoke out in favor of establishing a quota scheme to distribute migrants among European countries.
Her statement was condemned by the prime ministers of Hungary and Poland, Orbán and Szydło.
A. Merkel
V. Orbán
B. Szydło
Distribution of Migrants
Strict Migration Controls

Fig. 1. Discourse Network Example

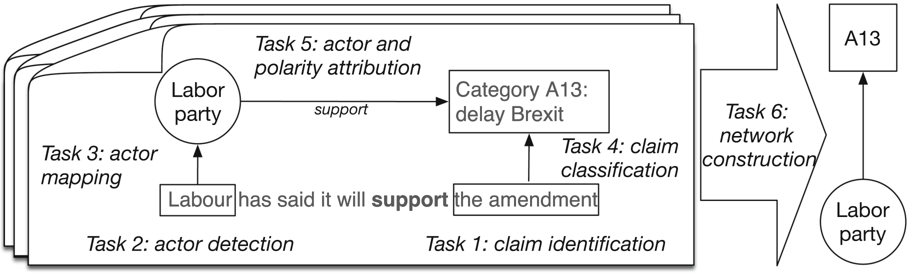

Task 5: actor and polarity attribution
Labor party
support
Category A13: delay Brexit
Task 3: actor mapping
Task 4: claim classification
Labour has said it will support the amendment
Task 2: actor detection
Task 1: claim identification
Task 6: network construction
A13
Labor party

Fig. 2. From newspaper articles to affiliation networks (adapted from [32])

graphs with two types of nodes, namely (a) actors (politicians, parties, organizations, but also groups of citizens such as protesters); and (b) fine-grained categories of claims (purposeful communicative acts in the public sphere by which an actor tries to influence a specific policy or political debate). Edges link actor nodes with the claim nodes that they communicate about and are tagged with a polarity: actors can either support or oppose specific claims. Figure 1 shows an example where actors are ovals, claim categories are rectangles, and green and red edges denote support and opposition. Figure 2 presents a step-by-step guide to developing such a network based on newspaper articles: Given a document, we need to detect text spans that express claims and actors (Tasks 1 and 2), we need to map these text spans onto canonical actors (e.g., “Merkel”, “the chancellor”, “Mrs. Merkel” are mentions of the canonical actor *Angela Merkel*) and claim categories, respectively (Tasks 3 and 4), and finally we need to establish actor-claim dyads with correct polarities (Task 5) and construct the actual network (Task 6). Until recently, to construct these networks, one needed to meticulously perform these tasks by hand; which costs time and hence money. Therefore, we aim to use NLP to develop predictive models capable of automating this process. This results in a fairly complex computational setup which gives rise to three main challenges:

(1) Annotation takes long and is costly. Traditional supervised learning demands a substantial number of annotated datapoints, but annotation of actors and claims calls for expert annotation. This leads to a ‘slow start’ situation: a sizable amount of manual annotation has to be carried out before computational modeling can proceed. Once models are in place, they can speed up future annotation, but this comes with its own set of challenges [18].

In practice, this means that a combination of time, money, and expertise is necessary to reach that point which might not always be available. Furthermore, carrying out comparative studies requires annotation to be available for multiple languages, even if only for evaluation purposes.

(2) Political claims are difficult to process on a fine-grained level. The codebooks developed by political science experts to describe the relevant claim categories in societal debates need to be sufficiently fine-grained to permit the characterization of competing positions in terms of the discourse network. This consideration often leads to codebooks with anywhere between 50 and over 100 claim categories [4,17,22]. As usual for language data, a few categories are frequent, while the majority are rare. This further exacerbates the problem mentioned in point (1) when learning claim identifications and claim classifiers (cf. Tasks 1 and 4 in Fig. 2): even a relatively large corpus will hardly provide enough examples of the infrequent categories for straightforward learning.

(3) Actor mentions are difficult to aggregate. Most of the mentions of actors in any discourse do not use their canonical name (“Angela Merkel”), but instead short versions (“Mrs. Merkel”), roles (“the chancellor”), or even just personal or possessive pronouns (“she”, “her”, compare Fig. 1). The mapping of such mentions onto the right actor node in the discourse network is essentially equivalent, in the general, to coreference resolution which is known to be a hard task. While shortcuts exist for some instances, notably the use of entity linking [36] for actors which are represented in some database, there are many actors for which this is not the case – including politicians at the local or regional levels as well as ‘ad-hoc’ actors such as “several ministers”.

In the following Sects. (2.1–2.4), we discuss a series of studies addressing tasks 1–4 from Fig. 2 and responding to these challenges. As gold standard for our studies we use DEbateNet [4], which is a large corpus resource that we created for the analysis of the German domestic debate on migration in 2015. After domain experts from political science developed a codebook for the policy domain, roughly 1000 newspaper articles from the German left-wing quality newspaper ‘taz - die tageszeigung’ with a total of over 550.000 tokens were annotated for actors, claims, and their relations, and finally used for computational modeling.

2.1 Less Annotation Is More: Few-Shot Claim Classification

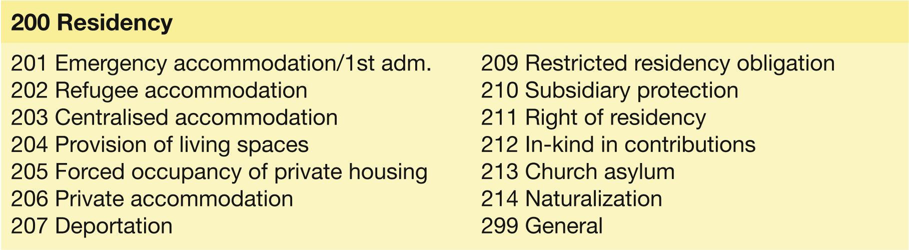
As noted in Challenge 1, NLP models that (partially) automate claim detection and classification traditionally require relatively large manually annotated data sets for training or fine-tuning, since the required domain-specific semantic distinctions are hard to recover directly from plain text. Since for most political topics no annotated data exists, research projects usually needed to start with a substantial amount of classical qualitative text analysis. The situation has changed substantially in the last two years, with the advent of large language models and their capacity for transfer learning and few-shot learning [5], that is, the ability to learn new tasks ad hoc, from very small numbers of examples.

To assess the potential of few-shot learning, we have carried out a study to assess whether we are able to replicate the manual annotation in one policy domain – the debate about the exit from nuclear energy in Germany in the year 2013 [19] – based on our models trained on migration debates and with a minimal amount of additional training data [17]. We thus try to process claims on the exit from nuclear energy use like “The Greens want to introduce a bill in the Bundestag for the immediate and final decommissioning of Germany’s seven oldest nuclear power plants” with a model trained on claims from the migration debate like “The basic right to asylum for politically persecuted persons knows no upper limit, Merkel also announced in an interview”. In this overview, we focus on the tasks of claim identification and claim classification (cf. Figure 2).

We work with a corpus of articles sampled with a keyword-based approach which still contains about as many relevant as irrelevant articles. Claims are identified by a binary sentence classifier. We start by calculating sentence embeddings using a sentence-BERT model (paraphrase-multilingual-mpnet-base-v2; [35]). We then use the manually annotated DEbateNet dataset (cf. Section 2) to train a multi-layer perceptron as claim identifier. Even though trained on data from a completely different topic area, our classifier obtains an F1 score of 0.78 on nuclear energy claims (precision: 0.77, recall: 0.79). This is remarkable, especially considering the large number of irrelevant articles in the corpus.

For claim classification, the model requires some information about the relevant claim categories. In our case, we use the category labels (i.e., names) from the codebook that was used to annotate the claims in the original study as minimal input for a few-shot learning approach. Again, each sentence is embedded with an SBERT model (using the same model as for claim identification). Analogously, the category labels from the codebook are encoded by SBERT. We then compute cosine similarity between all claim candidates and all category labels. Manually checking the top-ranked sentences for each label leads to seed sentences for each category. In the next step, we classify each claim candidate by assigning it to the category of the most similar seed sentence. To control the precision of claim classification, we introduce a threshold for similarity scores: Claim candidates with higher similarity scores are retained, while those below it are filtered out as potentially irrelevant.

When we evaluate whether the model correctly predicts categories for individual claims by computing F1 scores for each category, the model reaches F1 scores ranging from 0.23 to 0.45 for the more frequent claim categories (n > 20). Results for infrequent categories are unreliable. When evaluating whether the model correctly predicts the claims in the eight n-core networks of the original study, the results are better, with F1 scores between 0.29 and 0.69. In both cases, the variation of F1 scores across categories shows that especially infrequent categories pose a major challenge to our few-shot approach of generating discourse networks. In the next section, we therefore discuss options to increase the precision of predicting infrequent claim categories.



200 Residency	
201 Emergency accommodation/1st adm.	209 Restricted residency obligation
202 Refugee accommodation	210 Subsidiary protection
203 Centralised accommodation	211 Right of residency
204 Provision of living spaces	212 In-kind in contributions
205 Forced occupancy of private housing	213 Church asylum
206 Private accommodation	214 Naturalization
207 Deportation	299 General

Fig. 3. Codebook excerpt: Supercategory residency

Table 1. Claim classification: Precision, Recall, F-Scores on DebateNet newspaper corpus. Simplified from [13].

Freq band	Base			HLE			CRR			HLE+CRR		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Overall	61.2	41.9	47.0	75.2	52.2	59.0	70.4	49.0	55.2	76.5	54.3	**60.8**
Low	10.2	9.7	9.6	58.3	30.6	37.4	31.2	16.1	18.7	54.8	29.0	**35.8**
Mid	58.0	36.0	41.8	77.4	55.3	62.2	75.8	49.1	55.8	85.1	58.8	**66.2**
High	73.1	50.8	56.7	77.8	55.6	62.3	76.4	55.9	62.6	77.7	57.9	**64.0**

2.2 Improving Claim Classification with Hierarchical Information

As Challenge 2 formulated above, the many infrequent claim categories are very difficult to recognize accurately. We now describe a study on how to combat this challenge [13]. We start from the observation that political science codebooks are typically structured hierarchically into (at least) two levels, with an upper level that corresponds to broad political supercategories and a lower level that defines specific policies (claim categories). Figure 3 shows an example of the supercategory of residency, which is split up into about a dozen specific claim categories. When we have insufficient amounts of training data available, we can use this information to formulate a prior for the claim category embeddings that are learnt by our model's classifier: in the embedding space, claim categories should be located closer to other categories within the same supercategory than to claim categories of other supercategories. This leads us into the general area of hierarchy-aware classification methods.

While we evaluated various methods [13], we focus here on two approaches. The first one is hierarchical label encoding (HLE) [37] which decomposes the parameters for the specific claim categories into a (shared) supercategory part and a category-specific part, and a regularization approach which we call Class Representation Regularization (CRR) and which encourages the model to minimize the distances among the representations for each specific claim category within the same supercategory. For an experiment, we set up a base classifier based on BERT and combined it with HLE and CRR, training and testing on the DEbateNet gold standard (Sect. 2). The results in Table 1 shows that both

Table 2. Cross-lingual modeling: F1 scores for Claim class for claim identification (Id), macro average for claim classification (Class) on two datasets: DebateNet and Guardian. Simplified from [45].

Model	Train	Test	DebateNet		Guardian	
			Id F_1	Class F_1	Id F_1	Class F_1
Baseline (mono)	de	de	56.2	70.5	–	–
Translate-test	de	de (via en)	55.8	69.5	20.6	53.4
Translate-train	en	en	57.3	67.8	25.5	51.0
Multilingual	de	en	45.8	50.3	20.0	39.0

strategies, HLE and CRR, lead to a clear improvement in overall micro-F1 score over the base classifier (F_1=47.0) to F_1 scores of 59.0 and 55.2, respectively. A combination of the two leads to a further improvement to F_1=60.8. The improvement is most striking for the low frequency band (corresponding to the lowest frequency tercile), improving from F_1=9.6 to F_1=35.8. The developments for the two other frequency terciles are less dramatic, but still substantial (mid: 41.8 → 66.2, high: 56.7 → 64.0). A second study shows similar, albeit smaller, effects for claim classification on party manifestos with categories from MARPOR, a domain-independent claim classification schema [41] discussed in more detail below in Sect. 3.1. This bolsters the interpretation that our improvements are not tied to the specific codebook we used. We conclude that there is considerable space to improve the prediction quality for infrequent claim categories with dedicated methods.

2.3 Multilingual Claim Processing

When we move to another language while staying in the same policy domain – for example, for the purpose of comparative analyses across countries – we find ourselves faced with a specific case of Challenges 1 and 2: Do we have to start over with creating manual annotations? For argument mining, for which the identification of claims is a core task, the potential of machine translation for cross-lingual projection has already been established [16].

We report on a pilot study in claim identification and classification in other languages [45], machine-translating the German DEbateNet articles into English and French (this overview focuses on English). We compared three strategies: (a) backtranslating the foreign-language texts into German and analyzing them with a monolingual German BERT-based claim identifier and classifiers ('translate-test'); (b) building monolingual BERT-based foreign-language models from the translated DebateNet and using them to analyze the data in the respective languages ('translate-train'); (c) training multilingual models based on multilingual BERT on the original German data and then applying it to translated data ('multilingual').

The 'DebateNet' column of Table 2 shows that dealing with multilingual claims with machine translation works well: results are almost identical to the

Table 3. Actor mentions and their canonicalizations in newswire article (https://shorturl.at/WZ159)

	Local mention of actor	Canonical version
1	*President Joe Biden* pleaded with Republicans ...	Joe Biden
2	*Biden* signaled a willingness to make significant changes ...	Joe Biden
3	"We can't let Putin win", *he* said	Joe Biden

monolingual setup. In contrast, using multilingual embeddings incurs a substantial performance penalty. This is in line with previous analyses arguing that multilingual embeddings attempt to solve a harder, more open-ended task than MT systems do [2,34]. Also, claim identification in the multilingual embedding setup drops only ≈10 points F_1 compared to the baseline, while claim classification drops 20 points F_1 – the limiting factor seems to be the embeddings' (in-)ability to account for fine-grained topic distinctions consistently across languages.

This looks like machine translation is, indeed, sufficient to transfer political claims analysis across languages. However, the question is whether machine-translated text is a reasonable proxy for original text in a language. To test for this effect, we annotated a small sample of English reporting from the Guardian on the German migration debate. The results in the 'Guardian' column of Table 2 are much lower than those for the machine translated text. Again, we see an advantage for the MT-based approaches over multilingual embeddings, but less clearly. Particularly striking is the drop for claim identification with the MT approach from 56%-57% to 20-26% F_1. Indeed, a British newspaper is likely to report on German domestic affairs differently from a German newspaper, which leads to differences in claim form and substance: They tend to focus on the internationally most visible actors and report claims on a more coarse-grained level. Beyond the linguistic differences that NLP has so far focused on, therefore, working with newspaper reports from different countries necessitates bridging the cultural differences in framing [42], which may require some amount of manual labeling, or at least few-shot learning (cf. Section 2.1) after all.

2.4 Robust Actor Detection and Mapping

As outlined above, a central but difficult part of discourse network analysis is detecting actors for claims and mapping their textual mentions onto canonical forms (Tasks 2 and 3 in Fig. 2 and Table 3). We now describe a study comparing the two currently dominating approaches for this task [1]: (1) a pipeline of traditional NLP models, and (2) an end-to-end approach based on prompting a large language model (LLM). Once more, DebateNet, which provides a canonicalized representation for each actor, serves as dataset.

The pipeline approach comprises two steps. First, a CRF-based model identifies actor mention spans from the text, given the article with a marked claim as input. Since each claim has (at least) one actor, we constrain our CRF to always

Table 4. Prompt template instruction paraphrases used for robustness check for zero- and few-shot setting.

#	Instruction templates
1	*"Extract only the entity that made the claim in the article. The claim is surrounded with <claim> and <\claim> tags. Output only the entity without any additional explanation. Article: [ARTICLE]"*
2	*"Extract and standardize only the entity that made the marked claim in the article. The claim is surrounded with <claim> and <\claim> tags. Output only the standardized entity without any additional explanation. Article: [ARTICLE]"*
3	*"Retrieve the party or parties responsible for the statement in the given article, contained within <claim> and <\claim> tags. Output only the entity without further elaboration. Article: [ARTICLE]"*
4	*"Identify and output the entity or entities that made the claim within the specified article, enclosed by <claim> and <\claim> tags. Do not include any supplementary information. Article: [ARTICLE]"*

predict at least one actor mention per claim [33]. The second step of our pipeline canonicalizes these actors mentions through classification. We define the classes of this classifier to be (the string representations of) all canonicalized actors which appear at least twice in the training set (229, in our case), complemented by a special class 'keep-as-is' which covers all remaining actors mentions and which – true to its name – does not change the input. This heuristic approach works since infrequent actors are typically expressed by a linguistic expression that can serve well as a canonicalized version (either a full name, or a definition description such as 'the government secretaries'). The input to the classifier is the mention text and its article context. For both steps of our pipeline, we use a pre-trained XLM model [11] as an encoder, which we fine-tune during training.

In the LLM approach, we build on the pre-trained LLama 2 language model [40], directly predicting canonicalized actor strings as a text generation task, conditioned on a prompt containing the target claim. We compare zero- and few-shot prompting settings for base- and instruction-tuned model variants. For both settings, we construct the prompt following the current best practices [24,28,29]. The few-shot approach involves in-context learning, where the prompt contains a number of claim-actor pairs from the training set chosen by the cosine similarity score obtained from SBERT embeddings [35]. In our zero-shot approach, we do not include any claim-actor pairs in our prompt. Instead, we prompt our model with a short English-language description of the task. We experiment with various automatically constructed prompt paraphrases using ChatGPT shown in Table 4.

We evaluate our models via F_1-score. To better comprehend the strengths and weaknesses of both models, we use three evaluation settings. In our strictest *exact-match* setting, predictions are considered correct only if they exactly match the gold-standard actor string. *Correct-up-to-formatting* setting is more lenient

Table 5. Results for the LLM, traditional pipeline and hybrid models in the different evaluation settings.

	Evaluation	Pr	Re	F_1
LLM	exact match	42.66	43.46	43.06
	up to formatting	43.56	44.39	43.98
	up to canonic.	62.39	63.55	62.96
dedicated pipeline	exact match	48.66	59.35	53.47
	up to formatting	48.66	59.35	53.47
	up to canonic.	54.79	66.82	60.21
hybrid approach	exact match	54.33	64.49	58.97
	up to formatting	54.33	64.48	58.97
	up to canonic	64.96	79.39	70.21

by ignoring formatting differences (e.g. whitespaces, capitalization, punctuation) in the predictions. Lastly, in our *correct-up-to-canonicalization* setting, predictions are considered correct if they identify the correct entity, allowing variations in referring expressions For instance, both "the chancellor" and "Merkel" would be counted as correct predictions for the gold-standard actor "Angela Merkel".

Table 5 summarizes the results. While, under the strict exact-match setting, our traditional pipeline outperforms the LLM-based model, the LLM outperforms the pipeline when only evaluating up to canonicalization. This implies that the LLM is actually better than the pipeline at identifying the correct political actor, but struggles to canonicalize these actors consistently.

Motivated by this observation, we introduce a hybrid model that is structurally similar to our traditional pipeline model but includes the LLM prediction as an additional input. In this way, the pipeline can learn to delegate to the LLM when deciding which actor made the claim, and only has to properly canonicalize the LLM's prediction. Table 5 shows the hybrid model's performance under the same three evaluation settings. We find a substantial increase in performance across all settings. This suggests that our hybrid approach is able to leverage additional synergies between our two model architectures, improving upon the constituent models' abilities to both identify and canonicalize actors for claims.

3 Coarse-Grained Analysis of Political Discourse

We now proceed to the second approach, the analysis of manifestos to characterize parties. Party competition is a crucial mechanism in democracies. It creates an arena where a plurality of political viewpoints are given voice, enabling individuals to select one that aligns with their own beliefs. Analyzing this phenomenon is fundamental to understanding voters' choices during elections as well as the decisions taken by governing parties [3]. Researchers analyse party competition by, for example, placing them in a low-dimensional political space:

a one-dimensional left-right or libertarian-authoritarian, or conservative-liberal scale, or in a two-dimensional space formed by combining these scales [20].

We investigate the extent to which the positioning of parties can be captured through their manifestos – the electoral programs in which parties articulate their perspectives, plans, and objectives. Manifestos are crafted with the double intention of conveying information and persuading potential voters [8].

Political researchers analyze party manifestos to explore aspects such as level of similarity among parties concerning different policies [8], party alliances [15], and the alignment between voters' decisions with their worldviews [30]. By offering direct access to the parties' viewpoints, they serve as a robust foundation for comprehending the parties' ideologies regarding different policies. In contrast to the newspaper-based approach, manifesto-based analysis does not provide specific information about what types of decisions were made or articulated. On the other hand, it is arguably the most direct way of accessing the ideologies shared by members of the same party. It also avoids the filtering of information (via actors and their claims) through the lens of media.

3.1 Ideological Characterization

Traditionally, political science has approached the task of identifying party positioning by manually assigning a label to each sentence of a given manifesto. The Manifesto Project (MARPOR, [6]) is a well-known example that follows this method. Its annotations follow a codebook that classifies each sentence into a broader policy domain such as 'external relations' or 'freedom and democracy' as well as assigning a fine-grained label related to a specific category of the policy domain, such as 'freedom and human rights'. The category sometimes also encodes the stance, e.g., 'Constitutionalism: Positive or Negative'. These labels are then analyzed in terms of saliency, assuming that the most frequently mentioned policies are the most important ones for a party. A simplified version of saliency-based analysis is the RILE index, which is a coarse-grained measure that defines lists of 'left' and 'right' policies and simply calculates parties' position on the left-right scale as the relative mention frequency of left vs. right policies [7].

Manual annotations come with a substantial cost and must be carried out for each country and election. We ask whether we can alleviate this burden with unsupervised methods drawn from recent advancements in NLP. In [9], we empirically investigate the following questions with manifestos from Germany: 1) How to create embeddings for parties from their manifestos that yield robust between-party similarities estimates? 2) What aspects of document structure can be exploit for this purpose? 3) How well can these embeddings be computed in a completely unsupervised fashion?

We carry out experiments with six sentence embedding models, all of which estimate party positions on the basis of sentence similarity. These models range from a classic distributional model (fasttext) to transformers [35], applying whitening to ameliorate anisotropy [39] and comparing vanilla and fine-tuned versions. The results are shown in Table 6. Since we hypothesize that using only sentences expressing *claims* (cf. Section 2) might be more informative of the

Table 6. Correlation between our unsupervised scaling method and the ground truth (Wahl-o-Mat). Adapted from [9].

	Only claims		Entire manifestos	
Embeddings	**Domain**	**No domain**	**Domain**	**No domain**
fasttext$_{avg}$	*0.54	0.35	*0.44	0.41
BERT$_{german}$	0.37	*0.47	0.36	*0.48
RoBERTa$_{xml}$	0.39	*0.51	*0.46	*0.54
SBERT$_{vanilla}$	***0.57**	*0.50	***0.53**	*0.57
SBERT$_{domain}$	*0.44	*0.45	0.41	*0.52
SBERT$_{party}$	*0.53	***0.70**	*0.50	***0.69**

positioning of the parties, we introduce an experimental condition which considers only automatically identified claims (‘only claims’). Finally, we test whether computing sentence similarity within each MARPOR domain improves results (‘Domain’). See [9] for details.

We evaluate our unsupervised scaling method against similarities according to parties’ answers to the German Wahl-o-Mat questionnaire, a voting advice application (VAA) [43], generally considered a reliable estimation of party distances. Table 6 shows the results. The correlations are similar between the setup with *only claims* and *entire manifestos*, suggesting that claims are not much more informative than all sentences, at least when they are automatically recognized. The best model overall is based on fine-tuned SBERT$_{party}$ embeddings (which was fine-tuned to make statements by the same party more similar to one another) and computes similarities on an overall level instead of separately for each domain (column ‘No domain’). The lack of benefit from domain information might be surprising. One possible explanation is that voters prioritize different domains and do not simply ‘average’ across them [21].

We believe that these results hold promise for computational political science: leveraging document structure could potentially reduce the need for domain experts to annotate extensive amount of data. Our study has clear limitations, though: first, our experimentation was limited to a single language and dataset. In a follow-up study [31], we have established that classifiers based on state-of-the-art multilingual representations perform robustly in this task across countries and over time. Secondly, we have only considered a few document structure-based cues for fine-tuning. The range of available cues however is enormous and more research is needed in order to better understand the strengths and limitations of sentence embeddings.

3.2 Policy-Domain Characterization

The study from the previous subsection primarily considered the *aggregated* level of overall party positions [12,38]. Political scientists are, however, often interested in specific policy domains. We therefore ask, in [10], how well we can extend the approach presented above to the level of individual *policy domains.*

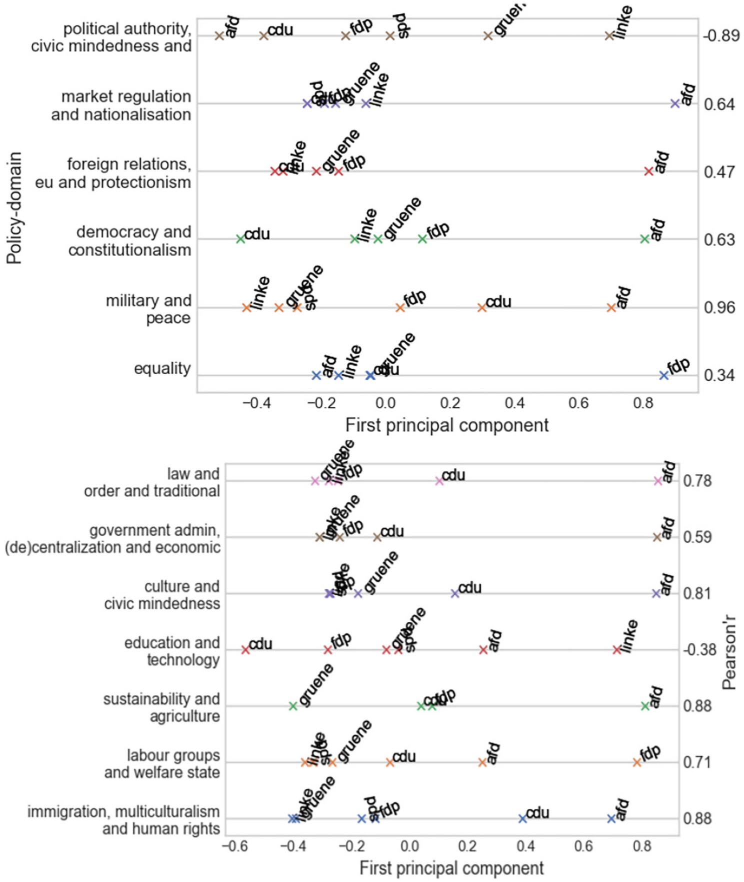


Fig. 4. Automatic prediction of German party positioning within policy domains (right-hand numbers: correlation with RILE scale).

Our approach computes distances between parties at the policy domain level by first training a *policy-domain labeller* which classifies the sentences of unannotated documents and then computing pairwise distances among sentences of the same policy domains across parties. We interpret the first principal component of the aggregated similarity matrix as a policy domain-specific scale.

Our experiments reveal that while the top-performing policy-domain labeller's accuracy is moderate (64.5%), the correlation between the predicted sentences and the ground truth – the RILE index (mentioned in 3.1) – remains remarkably high (r=0.79) in comparison with the annotated scenario (r=0.87). Figure 4 shows the positioning of parties per policy domain. In line with established observations about the German political landscape, a majority of policy domains exhibit a strong correlation to the RILE index, indicating a consistent adherence to the left-right scale. Where this is not the case (EU, market, government), a cluster of 'established' parties is clearly separated from the populist AfD. When evaluating the predicted setups against manual annotation, we find that the higher the accuracy of the policy-domain labeller within a class, the higher the correlation with the annotated results (Pearson r=0.59, p=0.03). This suggests that the accuracy of the labeller can be used as an indicator of which policy domains to reliably include in the analysis of unannotated manifestos.

This study verifies again that our methods perform well at an aggregated level of information by correlating highly with the RILE index. Moreover, our proposed workflow supplements the previous studies of party positioning with further detailed analysis within the sphere of policy domains. The predictions we obtain align closely with expert assessments, indicating that our workflow provides a reliable method to automatically compute the similarity between parties across some policy domains.

4 Conclusions

This paper considered the challenges of applying NLP methods for a text-based analysis of political debates. We compared two approaches: the first one aims at a fine-grained representation, taking individual statements (claims) and the political actors who made them as its building blocks, with the final goal of extracting discourse network representations from raw texts; the second one targets a coarse-grained representation of the debates at hand, with parties as the actors and their positions expressed in manifestos as its building blocks, with the final goal identifying global ideological positions, across languages and time.

As regards the fine-grained approach, our experiments and analyses show that current transformer-based language models have the potential to fundamentally change the way social scientists can use large text corpora to analyze political discourse. Whereas so far fine-grained analyses of political discourse have mostly been limited to short time spans, single countries or had to employ far-reaching sampling strategies to reduce the amount of texts to be processed. Following the pipeline from Fig. 1 we now know that *claim identification* (Task 1) needs to be preceded by a preparatory task to discard irrelevant documents, but after that, detection models work very well even on topics outside the original training data. *Actor detection* and *mapping* (Tasks 2 and 3) can be handled with reasonable success using traditional NLP methods such as entity extractors and classifiers respectively, but we also saw promising first results in using large language models to perform these two tasks jointly. However, owing to the inherent challenge in controlling the output generation of LLMs, the most effective strategy combines their capability to identify the correct actor and subsequently perform the canonicalization step within the traditional pipeline. For *claim classification* (Task 4), few-shot models show high potential, but they need human curation and re-calibration especially for infrequent claim categories.

While unable to fully automate annotation, current NLP tools go beyond just speeding up manual annotation processes. Topic agnostic claim detection models, few-shot learning, accounting for category hierarchies and models for actor mapping have the potential to restructure qualitative social sciences text analysis workflows. Instead of starting from zero with a small set of completely manually annotated texts, the current tools allow researchers to immediately focus on relevant text sections and potential claim sentences. With this a traditionally sequential annotation process can be replaced by a parallel and focused approach in which human interaction is mainly focusing on curation tasks.

When we turn to the coarse-grained approach, which aims at identifying the positioning of political parties based on manifestos, the verdict is even more optimistic. It shows performances similar to human annotators when identifying the positioning of parties in the well-established left-right scale (RILE index) or regarding their similarities according to Wahl-o-Mat. These results carry over, to an extent, to the level of individual policy domains – results for the annotated policy-domains correspond well to human expert judgements – but the task becomes considerably more difficult for the models. There is a clear need for further research on assessing the limits of the coarse-grained approach, and specifically on improving the performance of the classifier across policy-domains.

Thus, both approaches have advantages and disadvantages. The fine-grained discourse network analysis offers greater insights into what is being articulated in the public sphere and identifies the key political actors influencing or engaging in those discussions. However, even though we have shown that the annotation load can be alleviated with NLP tools, the task still requires extensive labelling, and it is very domain focused – i.e., each domain demands a new codebook and round of annotations. Besides that, the generalizations derived from the networks are dependent on what is reported by the media, where the focused claims and actors are selected by the news outlets. The coarse-grained approach based on manifestos, on the other hand, gives direct access to parties' policies and higher-level ideological positioning, reaching high quality with little to no annotations. That being said, the coarse-grained approach cannot provide detailed information about individual actors or claims in the political discourse, instead focusing on the relation among parties either at a policy-domain or at an ideological level.

Ultimately, we contend that the two approaches complement each other by offering distinct perspectives onto the political process: One illuminates the precise agreements and disagreements among actors, whereas the other offers an overview of party relatedness at a level of ideology or policy domains. Both offer insights and challenges that can be traded off according to the type of data, resources and analysis requirements at hand.

Limitations. The studies we presented in this paper were carried out primarily on newspaper text and party manifestos. While these are arguably two important text types for political discourse, they are by far not the only ones. Future work is necessary to determine the extent to which our findings carry over to other text types, notably oral modes such as (parliamentary) debates or intermediate modes such as social media communication. Similarly, the bulk of our work was concerned with German language texts. On the methodological perspective, it could take advantage of the relatively good NLP resource situation for German, leaving open the question of how to deal with similar situations in lower-resource languages. Our pilot studies [31,45] indicate that Machine Translation into a higher-resource language such as English appears a simple but effective solution for almost all languages at this point. At the data level instead, the annotations of German manifestos are recognized for their high quality due to the evaluation of inter-annotator agreement [25] – which may not be the case with manifestos from other countries. A crucial aspect to keep in mind are bias issues which could affect the models and thus result in unfair representations of the political discourse, i.e., overlooking actors from specific groups and/or their claims. While in [14] we have addressed frequency bias for claim detection (higher accuracy for claims by high frequency actors) a broader spectrum of unfairness sources is yet to be explored, in particular in the light of the use of LLMs.

Acknowledgments. The studies reported in this paper were funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through the projects MARDY-1 and MARDY-2 (grant number 375875969) within RATIO.

References

1. Barić, A., Padó, S., Papay, S.: Actor identification in discourse: a challenge for LLMs? In: Proceedings of the EACL CoDi Workshop, pp. 64–70. St. Julians, Malta (2024)
2. Barnes, J., Klinger, R.: Embedding projection for targeted cross-lingual sentiment: model comparisons and a real-world study. JAIR **66**, 691–742 (2019)
3. Benoit, K., Laver, M.: Party Policy in Modern Democracies. Routledge (2006)
4. Blokker, N., Blessing, A., Dayanik, E., Kuhn, J., Padó, S., Lapesa, G.: Between welcome culture and border fence: the European refugee crisis in German newspaper reports. LRE **57**, 121–153 (2023)
5. Brown, T., et al.: Language models are few-shot learners. In: Proceedings of NeurIPS, pp. 1877–1901 (2020)
6. Budge, I.: Validating the manifesto research group approach: theoretical assumptions and empirical confirmations. In: Laver, M. (ed.) Estimating the Policy Position of Political Actors, pp. 70–85. Routledge (2001)
7. Budge, I.: The standard Right–Left scale. Technical report, Comparative Manifesto Project (2013). https://manifesto-project.wzb.eu/down/papers/budge_right-left-scale.pdf
8. Budge, I., Klingemann, H.D., Volkens, A., Bara, J., Tanenbaum, E. (eds.): Mapping Policy Preferences: Estimates for Parties, Electors, and Governments 1945–1998. OUP, Oxford, New York (2001)
9. Ceron, T., Blokker, N., Padó, S.: Optimizing text representations to capture (DIS)similarity between political parties. In: Proceedings of CoNLL. Abu Dhabi, UAE (2022)
10. Ceron, T., Nikolaev, D., Padó, S.: Additive manifesto decomposition: a policy domain aware method for understanding party positioning. In: Findings of ACL. Toronto, Canada (2023)
11. Conneau, A., Lample, G.: Cross-lingual language model pretraining. In: Proceedings of NeurIPS, Vancouver, Canada (2019)

12. Däubler, T., Benoit, K.: Scaling hand-coded political texts to learn more about left-right policy content. Party Politics 13540688211026076 (2021)
13. Dayanik, E., et al.: Improving neural political statement classification with class hierarchical information. In: Findings of ACL, Dublin, Ireland (2022)
14. Dayanik, E., Padó, S.: Masking actor information leads to fairer political claims detection. In: Proceedings of ACL, pp. 4385–4391. Online (2020)
15. Druckman, J.N., Martin, L.W., Thies, M.F.: Influence without confidence: upper chambers and government formation. LSQ **30**(4), 529–548 (2005)
16. Eger, S., Daxenberger, J., Stab, C., Gurevych, I.: Cross-lingual argumentation mining: machine translation (and a bit of projection) is all you need! In: Proceedings of COLING. Santa Fe, New Mexico, USA (2018)
17. Haunss, S., Blessing, A.: Revisiting the exit from nuclear energy in Germany with NLP. Zeitschrift für Diskursforschung (2023, under review)
18. Haunss, S., Blokker, N., Blessing, A., Dayanik, E., Lapesa, G., Kuhn, J., Padó, S.: Integrating manual and automatic annotation for the creation of discourse network data sets. Politics Govern. **8**(2), 326–339 (2020)
19. Haunss, S., Dietz, M., Nullmeier, F.: Der Ausstieg aus der Atomenergie. Diskursnetzwerkanalyse als Beitrag zur Erklärung einer radikalen Politikwende. Zeitschrift für Diskursforschung **1**(3), 288–316 (2013)
20. Heywood, A.: Political Ideologies: An Introduction. Bloomsbury Publishing (2021)
21. Iversen, T.: Political leadership and representation in West European democracies: a test of three models of voting. AJPS **38**(1), 45–74 (1994)
22. Kammerer, M., Ingold, K.: Actors and issues in climate change policy: the maturation of a policy discourse in the national and international context. Soc. Netw. **75**, 65–77 (2023)
23. Koopmans, R., Statham, P.: Political claims analysis: integrating protest event and political discourse approaches. Mobilization **4**(2), 203–221 (1999)
24. Kumar, S., Talukdar, P.: Reordering examples helps during priming-based few-shot learning. In: Findings of ACL-IJCNLP, pp. 4507–4518. Online (2021)
25. Lacewell, O.P., Werner, A.: Coder training: Key to enhancing coding reliability and estimate validity. Mapping Policy Preferences from Texts, Statistical Solutions for Manifesto Analysts (2013)
26. Leifeld, P.: Discourse network analysis: policy debates as dynamic networks. In: Victor, J.N., Montgomery, A.H., Lubell, M. (eds.) The Oxford Handbook of Political Networks. OUP (2016)
27. Leifeld, P., Haunss, S.: Political discourse networks and the conflict over software patents in Europe. Eur. J. Polit. Res. **51**(3), 382–409 (2012)
28. Lu, Y., Bartolo, M., Moore, A., Riedel, S., Stenetorp, P.: Fantastically ordered prompts and where to find them: overcoming few-shot prompt order sensitivity. In: Proceedings of ACL, pp. 8086–8098. Dublin, Ireland (2022)
29. Margatina, K., Schick, T., Aletras, N., Dwivedi-Yu, J.: Active learning principles for in-context learning with large language models. In: Findings of EMNLP, pp. 5011–5034. Singapore (2023)
30. McGregor, R.M.: Measuring "correct voting" using comparative manifestos project data. J. Elect. Publ. Opinion Parties **23**(1), 1–26 (2013)
31. Nikolaev, D., Ceron, T., Padó, S.: Multilingual estimation of political party positioning: from label aggregation to long-input transformers. In: Proceedings of EMNLP. Singapore (2023)
32. Padó, S., Blessing, A., Blokker, N., Dayanik, E., Haunss, S., Kuhn, J.: Who sides with whom? Towards computational construction of discourse networks for political debates. In: Proceedings of ACL. Florence, Italy (2019)

33. Papay, S., Klinger, R., Padó, S.: Constraining linear-chain CRFs to regular languages. In: Proceedings of ICLR. Online (2022)
34. Pires, T., Schlinger, E., Garrette, D.: How multilingual is multilingual BERT? In: Proceedings of ACL. Florence, Italy (2019)
35. Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: Inui, K., Jiang, J., Ng, V., Wan, X. (eds.) Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 3982–3992. Association for Computational Linguistics, Hong Kong, China (November 2019). https://doi.org/10.18653/v1/D19-1410, https://aclanthology.org/D19-1410
36. Sevgili, O., Shelmanov, A., Arkhipov, M., Panchenko, A., Biemann, C.: Neural entity linking: a survey of models based on deep learning. Semant. Web **13**(3), 527–570 (2022)
37. Shimaoka, S., Stenetorp, P., Inui, K., Riedel, S.: Neural architectures for fine-grained entity type classification. In: Proceedings EACL. Valencia, Spain (2017)
38. Slapin, J.B., Proksch, S.O.: A scaling model for estimating time-series party positions from texts. Am. J. Polit. Sci. **52**(3), 705–722 (2008)
39. Su, J., Cao, J., Liu, W., Ou, Y.: Whitening sentence representations for better semantics and faster retrieval. arXiv:abs/2103.15316 (2021)
40. Touvron, H., et al.: Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288 (2023)
41. Volkens, A., et al.: The Manifesto data collection, version 2019b (2019)
42. Vu, H.T., Liu, Y., Tran, D.V.: Nationalizing a global phenomenon: a study of how the press in 45 countries and territories portrays climate change. Glob. Environ. Chang. **58**, 101942 (2019)
43. Wagner, M., Ruusuvirta, O.: Matching voters to parties: voting advice applications and models of party choice. Acta Politica **47**(4), 400–422 (2012)
44. de Wilde, P.: No polity for old politics? A framework for analyzing the politicization of European integration. J. Eur. Integr. **33**(5), 559–575 (2011)
45. Zaberer, U., Padó, S., Lapesa, G.: Political claim identification and categorization in a multilingual setting: first experiments. In: Proceedings of KONVENS. Ingolstadt, Germany (2023)
46. Zürn, M.: The politicization of world politics and its effects: eight propositions. EPSR **6**(01), 47–71 (2014)

PAKT: Perspectivized Argumentation Knowledge Graph and Tool for Deliberation Analysis 🤝

Moritz Plenz[1](✉), Philipp Heinisch[2], Anette Frank[1], and Philipp Cimiano[2]

[1] Department of Computational Linguistics, Heidelberg University, Heidelberg, Germany
{plenz,frank}@uni-heidelberg.de
[2] CITEC, Bielefeld University, Bielefeld, Germany
{pheinisch,cimiano}@techfak.uni-bielefeld.de

Abstract. Deliberative processes play a vital role in shaping opinions, decisions and policies in our society. In contrast to persuasive debates, deliberation aims to foster understanding of conflicting perspectives among interested parties. The exchange of arguments in deliberation serves to elucidate viewpoints, to raise awareness of conflicting interests, and to finally converge on a resolution. To better understand and analyze the underlying processes of deliberation, we propose PAKT, a Perspectivized Argumentation Knowledge Graph and Tool. The graph structures the argumentative space across diverse topics, where arguments i) are divided into premises and conclusions, ii) are annotated for stances, framings and their underlying values and iii) are connected to background knowledge. We show how to construct PAKT and conduct case studies on the obtained multifaceted argumentation graph. Our findings show the analytical potential offered by our framework, highlighting the capability to go beyond individual arguments and to reveal structural patterns in the way participants and stakeholders argue in a debate. The overarching goal of our work is to facilitate constructive discourse and informed decision making as a special form of argumentation. We offer public access to PAKT and its rich capabilities to support analytics, visualization, navigation and efficient search, for diverse forms of argumentation (GitHub: www.github.com/Heidelberg-NLP/PAKT Website: www.webtentacle1.techfak.uni-bielefeld.de/accept/).

Keywords: Argumentation · Deliberation · Knowledge Graph

1 Introduction

Deliberative processes play a vital role in shaping opinions, decisions, and policies in society. Deliberation is the collaborative process of discussing contested issues, to collect and form opinions and guide judgment, in order to find consensus among stakeholders. The key underlying idea is that groups are able to make

M. Plenz and P. Heinisch—The authors contributed equally.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 89–107, 2024.
https://doi.org/10.1007/978-3-031-63536-6_6

better decisions regarding societal problems than individuals.[1] Deliberation thus can change minds and attitudes, provided that participating individuals are willing to communicate, advocate and to become persuaded with and by others [24]. Effective deliberation, whether in person or online, incorporates sustained and sound modes of argumentation [10] and can take many forms: from (moderated) discussions to role-playing or formal debates. All these activities aim to explore differing perspectives and should lead to informed and inclusive decisions.

Deliberative theory is concerned with investigating and theorizing about how people discuss and come to conclusions. It has been argued that public debates as available in online debating or discussion fora, or social media platforms such as Reddit, are black boxes, as we have little knowledge about how people argue and what their arguments are based upon [24]. Thus, effective tools are needed to shed light on existing debates to better understand how people argue.

In this work we propose a new framework to support advanced analytics of argumentative discourse, which we apply to analyze deliberative discussions, as a special form of argumentation. At the core of our framework is PAKT, a *Perspectivized Argumentation Knowledge Graph and Tool* that relies on a data model suited to formalize and connect argumentative discussions – be it interactive dialogues or exchanges in Web fora – enabling a multi-dimensional analysis of the content of arguments, their underlying perspectives and values, and their connection to different stakeholder groups and to background knowledge. PAKT builds on the theory of argumentation by segmenting arguments into premises and conclusions, and focuses on their perspectivization by specifying frames and values which arguments highlight or are based on, and using knowledge graphs to ground arguments in relevant background knowledge.

By going beyond single arguments, PAKT characterizes debates at a structural level, revealing patterns in the way specific groups of stakeholders argue and allowing us to analyze important quality aspects of deliberative discussions. Hence, PAKT aids in understanding *how people argue*, including question such as i) *Given a debated issue, are (all) relevant argumentative perspectives covered?* ii) *Who provided which argument(s)?* and *What are common framings, underlying values and perspectives in presenting them?* and iii) *How do these perspectives and values differ between pro and con sides, and stakeholder groups?*

We leverage and refine state-of-the-art argument mining and knowledge graph construction methods to build a rich, perspectivized argumentation knowledge graph, by applying them to debates from `debate.org` (DDO) as a proof of concept. We show how to analyze this graph in view of its underlying model, and how to answer the above questions by applying PAKT as an analytical tool.

Our main contributions are: We i) introduce PAKT, a framework for deliberation analysis that we ii) apply to `debate.org` as a proof of concept. We iii) demonstrate how to use it to examine deliberative processes, and iv) offer case studies that leverage PAKT to analyze debates from a deliberative viewpoint.

[1] Cf. Habermas, Cohen, Dryzek, Fishkin, see https://tinyurl.com/2p9vsha7.

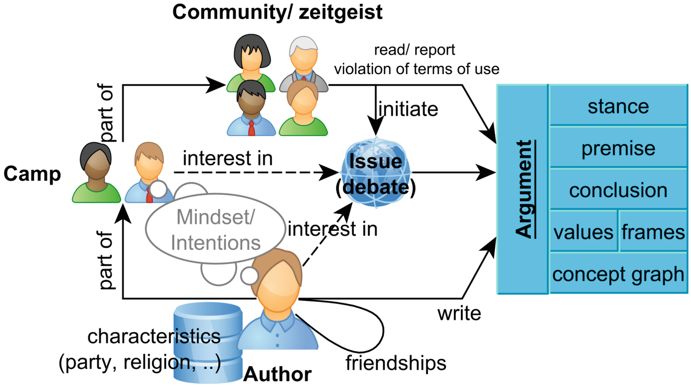


Fig. 1. PAKT data model consisting of arguments (w/ premises, conclusions, frames, values, stance towards topic and concepts) and authors, camps, zeitgeist

2 A Data Model for Perspectivized Argumentation

Debates in the real world are fundamentally driven by the **interaction of individuals**. These individuals play various roles in a debate, such as *authors* or members of the *audience*, each bringing unique values, preferred framings and areas of interest into discussions. The *individual characteristics of participants* clearly influence the arguments they formulate and those they engage with.

To unravel the complex interplay between individuals and arguments in real-world debates, we present a **human-centered model** (Fig. 1) of a perspectivized argumentation knowledge graph which serves as a structured framework for capturing dynamics in argumentation. Through this formalization, we aim to shed light on the intricacies of framed argumentation, to enhance our understanding of how individuals engage in discussion, and how they can help shaping the quality and outcome of debates, to make them *deliberative*.

Authors, as all individuals, have diverse *beliefs*, *values* and *issues of interest*. Individuals who share properties naturally coalesce into **camps**, which may manifest as formal entities, e.g., political parties, or informal gatherings. Importantly, camps need not adhere to formal memberships, and individuals can participate in multiple camps, even if they hold partially contradictory positions.

By uniting all individuals or camps within a **community**, we arrive at the concept of the *zeitgeist*-a collective repository of beliefs and norms. It governs the relevance and controversy of issues, and thereby shapes the landscape of debates. It also influences the arguments presented within these debates. Arguments that violate the code of conduct, e.g., are typically avoided by authors or moderated out. *Readers*, being part of the community, assess arguments through the lens of the zeitgeist, which can impact their agreement or conviction levels.

Authors, guided by personal convictions or their camps' interests, craft **arguments** on specific issues. Arguments usually comprise a *premise* and *conclusion*, and reflect a particular *stance* on the issue at hand. Arguments reveal additional information by exposing specific *framings*, *values*, or *concepts* that authors (often deliberately) use to convey their message. Note that these choices can be influenced by the author, their camps, the zeitgeist, or even the audience.

A **debate** is formed by all arguments on a specific **issue** put forth by its participants. A good *deliberative* debate should cover all relevant aspects of the issue. This can be achieved by including all *interested parties* and by exploring *(counter-)arguments* of all stances that consider different perspectives and viewpoints of individuals and camps, while ensuring the soundness of each argument.

3 Constructing PAKT$_{DDO}$ from `debate.org`

This section describes, as proof of concept, how we apply PAKT to represent debates from `debate.org` (DDO for short) and which methods we apply to construct the graph. Minor implementation details are in our supplementary materials [21].

3.1 Arguments from `debate.org`

Figure 1 shows two core components of PAKT: i) a *set of arguments* discussing debatable issues and ii) *authors of these arguments*, who can be related to each other. While existing argumentative datasets [1,16,33] do not include author information, a well-known platform that hosts a rich source of arguments along with author profiles is the former debate portal `debate.org` (DDO).[2] This debate portal has been crawled and used in the field of argument mining several times [7,8,34]. To further broaden the extracted data of this portal, we selected 140 controversial issues with at least 25 contributed opinions each, yielding overall 24,646 arguments, where a user profile is available for 7,001 arguments.

Stance, Premise and Conclusion of Arguments. The DDO portal presents controversial issues as questions that users answer with *yes* (pro) or *no* (con), followed by a *header* and a *statement* (opinion) that explains the answer in detail. We construct arguments from this data by interpreting the provided statement as the *premise* and automatically generating a *conclusion*. Consider the example:

Issue	Should animal hunting be banned?
Stance	pro
Header	Sport hunting should be banned
Statement	"[...] Hunting for fun or sport should be banned. How is it fun killing a defenseless animal that's harming no one? [...]"

Conclusion Generation. Since conclusions are not given in the DDO data, we construct conclusions automatically. For this we apply ChatGPT in a few-shot setting, showing it three examples consisting of i) the question, ii) stance, iii) header, and iv) a manually created conclusion. For our example, the generated conclusion is "*Sport hunting should be banned in order to protect animals.*" The complete prompt is shown in our supplementary materials [21].

[2] The website went offline on 5th of June, 2022. See Fig. 5 for an example screenshot.

3.2 Characterizing Arguments for Perspectivized Argumentation

We enrich arguments with automatically inferred frames, values and concept graphs to enable easy analysis and filtering in PAKT.

Frames. To represent specific viewpoints, perspectives, or aspects from which an argument is made, we adopt the notion of "frames." While one line of research tailors frame sets to each issue separately, yielding *issue-specific* frame sets [1, 27, 28], we aim to generalize frames across diverse issues. We therefore apply the MediaFrames-Set [5], a *generic* frame set consisting of 15 classes that are applicable across many issues and topics.

To apply these frames to arguments from DDO, we fine-tune a range of classifiers on a comprehensive training dataset of more than 10,000 newspaper articles that discuss immigration, same-sex marriage, and marijuana, containing 146,001 text spans labeled with a single MediaFrame-class per annotator [6]. To apply this dataset to our argumentative domain, we broaden the annotated spans to sentence level [13]. Since an argument can address more than a single frame [26], we design the argument-frame classification task as a multi-label problem by combining all annotations for a sentence into a frame target set. To introduce additional samples with more comprehensive text and target frame sets, we merge existing samples pairwise by combining their text and unifying their target frame set. As processing architecture, we apply different architectures [14], and determine LLMs (RoBERTa [19][3]) as the best-performing ones.

Human Values. Since we aim to analyze arguments not as standalone text, but as text written by individuals with intentions and goals, it is also important to analyze the human values [2, 15, 17, 36] underlying a given argument, to infer the authors' beliefs, desirable qualities, and general action paradigms [15]. The shared task "SemEval 2023 Task 4: ValueEval" [16] popularized the Schwartz' value continuum [30]. This is a hierarchical system with four higher-order categories: "Openness to change", "Self-enhancement", "Conversation", and "Self-transcendence". At the second level, these categories are refined into 12 categories, including "Self-direction", "Power", "Security", or "Universalism". To reduce the complexity of the value classification task, we follow Kiesel *et al.* [16] in not using the finest granularity of Schwartz' value continuum, but rather the second-smallest level containing 20 classes. For predicting value classes for an argument, we rely on a fine-tuned ensemble of three LLMs published by the winning team [29] of the shared task.

Concepts. Humans possess rich commonsense knowledge that allows them to communicate efficiently, by leaving information implicit that can be easily inferred in communication by other humans. Also in argumentation, it is often left implicit how a conclusion follows from a given premise. To uncover which concepts are covered in a given argument – either explicitly or implicitly – we link arguments to ConceptNet [32], a popular commonsense knowledge graph.

[3] For further studies in this paper, we apply the checkpoint https://huggingface.co/pheinisch/MediaFrame-Roberta-recall.

To do this we rely on [22] to extract subgraphs from ConceptNet: We split the premise into individual sentences (cf. [14]), then, for each sentence in the premise and for the conclusion, we extract relevant ConceptNet concepts. These concepts represent explicit mentions in the premise and conclusion, but not implicit connections. Hence, we connect the extracted concepts with weighted shortest paths extracted from ConceptNet. These paths reveal how the conclusion follows from the premise, along with other potential implicit connections [22].

3.3 Authors and Camps

In DDO, authors could choose to reveal their user profile when posting an argument. To model stakeholder groups, we group users into camps using their user profiles. The profiles state distinct categories for traits such as *gender*, *ideology*, *religion*, *income*, or *education*. Users could also fill free-text fields about, e.g., personal beliefs or quotes. Users control which parts of their profiles are public, so the amount of available data differs for each user. To obtain camps, we cluster the stated categories in coarse groups, e.g. *left*, *right* and *unknown* for ideology.

3.4 Implementation and Tools for Building and Using PAKT

PAKT is designed to aid in future argumentative analysis, so we make it publicly available in several forms. Our website[4] provides a comprehensive overview of issues in PAKT_{DDO} in a search interface. To enable richer analysis we also make PAKT_{DDO} available as a Neo4J[5] graph database that loosely follows the structure shown in Fig. 1. Neo4J databases can be queried with *Cypher*, a powerful, yet easy-to-learn querying language similar to SQL, but that supports queries on graphs. Issues, users, arguments, and other entities can efficiently be searched for and filtered in our database. A detailed description on how to utilize our database can be found at www.github.com/Heidelberg-NLP/PAKT.

3.5 Preliminary Evaluation

To provide a preliminary evaluation of the quality of the PAKT_{DDO} graph, we manually labeled 99 arguments on the issue "*Should animal hunting be banned?*" that will be used in our case study (Sect. 5.1). We evaluate the quality of generated conclusions and annotated labels (frames and values), as well as retrieved supporting and counter arguments. Each annotation sample includes the stance, the header, and the full statement (premise). For each argument, three annotators provided judgments on five questions[6]: (i) *Conclusion quality* (rating the appropriateness of the conclusion generated by ChatGPT): 94/99 conclusions are labeled as appropriate; (ii) *Frame identification* (identifying all emphasized

[4] https://webtentacle1.techfak.uni-bielefeld.de/accept.

[5] https://neo4j.com.

[6] The labels were aggregated using the majority vote.

aspects): the predictions yield 0.40 micro-F1; (iii) *Human value detection* (detecting all values encouraged by the argument): again the predictions yield 0.40 micro-F1; (iv) *Similarity rating* (given two further arguments, rating whether and which argument is more similar): similarity predictions for arguments with the same stance obtained with S^{3}BERT [20] correlate with annotator judgments with an accuracy of 42%; (v) *Counter rating* (given two further arguments, rating whether and which arguments attack the given argument more): the similarity predictions for arguments with the opposite stance obtained from S^{3}BERT [20] correlate with an accuracy of 40%. For detailed analysis of the manual study including IAA see our supplementary materials [21].

4 Analytics Applied to PAKT$_{DDO}$

In this section we analyse PAKT$_{\mathrm{DDO}}$ at a global level to discover general trends in our data, by aggregating information across all represented issues.

Frames and Values. Figure 2 (left) shows the distribution of frames and human values across all arguments from all issues. The frames *health and safety*, *cultural identity*, *morality* and *quality of life* are the most frequent, each occurring in almost 20% of all arguments. The most common values are *concern* (49%) and *objectivity* (45%). We further observe that some frames occur frequently with certain values and vice versa. The *fairness and equality* frame, e.g., occurs six out of seven times in combination with the value *concern.*

Concepts. For our analysis in this paper, we consider the ratio of arguments that mention a certain concept. To avoid biases due to the structural properties of ConceptNet (e.g. some concepts are better connected and hence occur more often), we report these ratios relative to the ratio computed over all arguments in PAKT$_{\mathrm{DDO}}$. E.g., when reporting the concept ratios for a specific frame, we report the ratio relative to the ratio computed over all arguments that we subtract from the former, i.e., $\frac{N_{fc}}{N_f} - \frac{N_c}{N}$, where N is the number of arguments with a specific frame f or concept c. When comparing two subsets of PAKT$_{\mathrm{DDO}}$ – for example pro and con on a certain topic – we instead normalize by the complementary subset to obtain more specific concepts.

When linking arguments to commonsense background knowledge we see that the most frequent concepts are *Person* and *People*, indicating that most debates are – as expected – human-centered. Other commonly occurring concepts are *US*, *Legal*, *War*, or *School* which reflect the categories and context that our issues stem from. These concepts are also frequently used in contemporary debates, which indicates that issues in PAKT$_{\mathrm{DDO}}$ are representative for general debates.

Our analysis also reveals concepts that are specific to certain frames and values. For example, the concepts *religion*, *god*, *person*, *biology*, *human* and *christianity* occur between 10 and 24% points (pp) more often in arguments bearing the *morality* frame, compared to all arguments across all frames. Similarly, for the value *nature*, the most common concepts are *animals*, *animal*, *zoo*, *kept in*

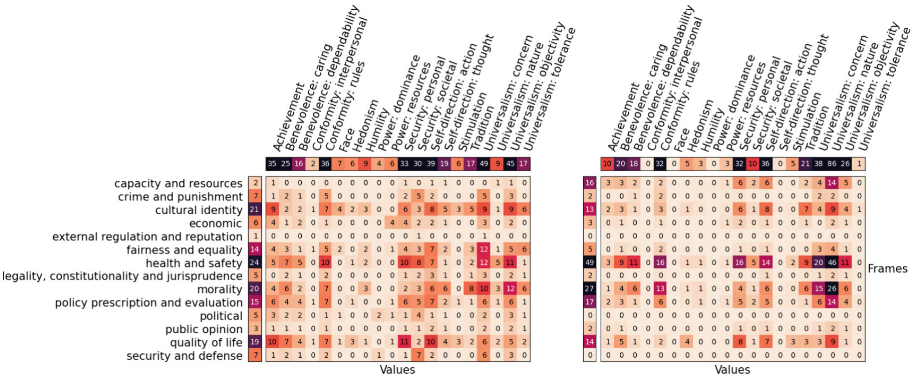


Fig. 2. Correlation between frames and values. Left plot is across all topics, right plot is for the issue *Should animal hunting be banned?* Arguments labeled with more than one frame/value are counted multiple times. Numbers are percentages.

zoos, *killing* and *water*, which occur between 12 and 39 pp more often than in all arguments.

Camps. PAKT$_{\text{DDO}}$ includes author information that users have decided to provide for themselves. Using this information, we can group users (i.e. the authors of arguments) into camps along several dimensions, as described in Sect. 3.3. This allows us to compare which frames and values are preferred by which camps. Figure 3a shows these distribution for authors of different ideology. In comparison, left-winged authors prefer the *objectivity* and *self-direction: action* values, while right-winged authors consider the values *tradition* and *conformity: rules* more. For frames, the difference between the camps is relatively small, indicating that one's ideology is more value-driven. Figure 7 shows the distributions for other camps, where we observe stronger effects for frames.

However, since different issues have different relevance for single frames and values, we check whether different distributions of frames and values are caused by different issue participation dependent on the camp. Here, our analysis shows that authors from different camps engage in issues from similar categories, with participation rates differing by at most ∼3 pp for ideology (cf. Fig. 6), showing that different camps prefer different frames and values while debating on the same issues.

5 Case Studies

5.1 Should Animal Hunting Be Banned?

For deeper analysis we examine one specific issue, namely *Should animal hunting be banned?* PAKT$_{\text{DDO}}$ contains 409 arguments on this issue, with a relatively even parity (∼46% pro and 54% con).

Camps. Our notion of camps used in Sect. 4 requires user information, which is scarce at the level of individual issues. For example, for ideology only 17 contributing authors provided user information. Therefore, for the given issue we consider people in favor and against banning animal hunting as distinct camps. Separating authors into camps by their stance actually does reflect the friendship network between authors on DDO, as shown in Fig. 4.

Frames and Values. Figure 2 (right) shows the frames and values for this issue. 86% of arguments address the *nature* value, which is directly linked to the issue. Other frequent values occurring in more than 30% of arguments are *universalism: concern*, *self-direction: action*, *conformity: rules* and *security: personal.* The most frequent frames are *health and safety* and *morality.*

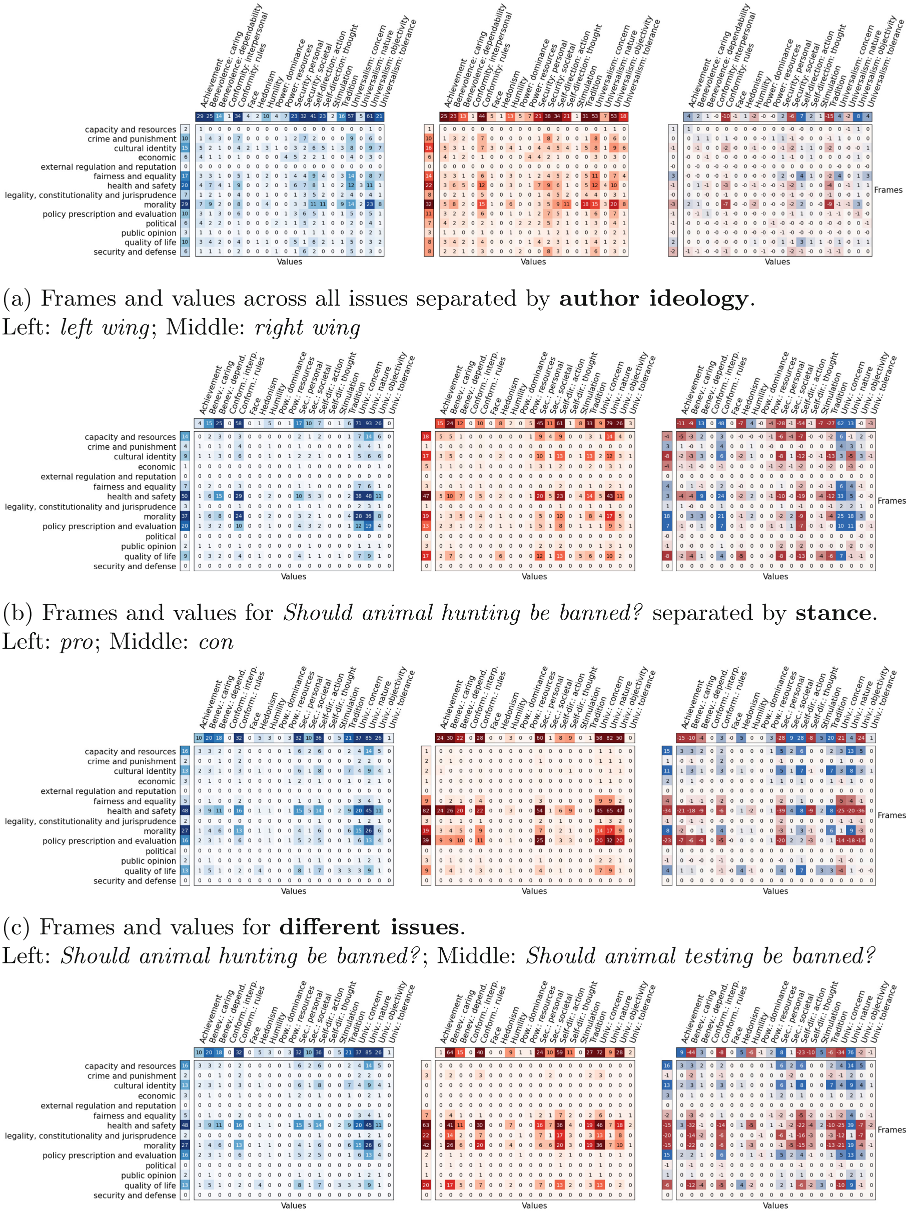
To better understand how and why these frames and values arise, we look at how they differ between stances (Fig. 3b). Firstly, we note that the most frequently occurring frames and values are common for both stances. However, manual inspection of these arguments reveals that these frames and values are interpreted in different ways. For example, on the pro side the *nature* value often refers to species or entire ecosystems being endangered, and that humans should not diminish them even more. By contrast, on the con side, a common interpretation of nature protection is that balance needs to be maintained by hunting over-populating species such as deer. Identifying such shared values with different interpretations can aid in finding common ground and ultimately satisfying compromises. Here, a possible compromise could be to ban the hunting of endangered species, but to allow sustainable hunting of certain species.

However, a value or frame can also predominantly be used by a certain stance. The value *universalism: concern* expresses that all people and animals deserve equality, justice, and protection. 71% of all pro arguments support this value, while only 9% of all con arguments support it. On the pro side, this value means that we shouldn't hunt animals, as we also would not hunt humans. Authors on the con side addressing this value argue that hunted animals have better lives than farmed animals. Again, the difference lies in the interpretation.

Concepts. For our target issue, we obtain concepts revolving around animals, hunting, killing, and food. Again, we compare pro and con arguments to each other: The most prominent pro-concepts are *killing animal*, *killing*, *bullet*, *animals*, *evil* and *stabbing to death.* On the other hand, the most frequently occurring con-concepts are *getting food*, *fishing*, *eat*, *going fishing*, *meat* and *food.* This highlights the different foci regarding hunting: people in favor of banning hunting emphasize the aspect of killing during hunting, while people who oppose a ban on hunting emphasize the usage of dead animals for food. Hence, the concepts can be seen as issue-specific framings used by the pro and con sides.

5.2 Comparison to Other Issues

An important aspect of opinion-making, and hence of deliberation, is to learn from similar debates. Similar issues can be identified with standard similarity prediction methods like SBERT [20,25], which is already integrated in PAKT.



(a) Frames and values across all issues separated by **author ideology**. Left: *left wing*; Middle: *right wing*

(b) Frames and values for *Should animal hunting be banned?* separated by **stance**. Left: *pro*; Middle: *con*

(c) Frames and values for **different issues**. Left: *Should animal hunting be banned?*; Middle: *Should animal testing be banned?*

(d) Frames and values for **different issues**. Left: *Should animal hunting be banned?*; Middle: *Should Abortion be illegal in America?*

Fig. 3. Comparison between frame and value matrices. The left and middle plots show distributions in percent, and the right plots show their differences in percentage points (pp).

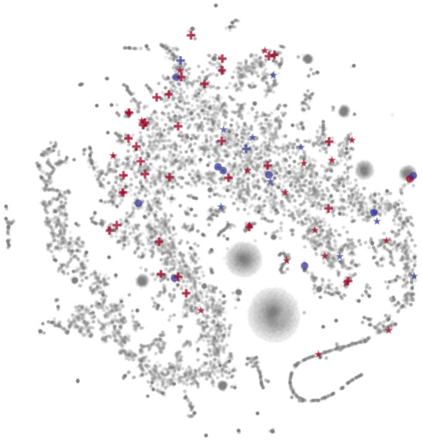


Fig. 4. T-SNE embedding of the spectral embeddings of the largest connected component of the friendship network of DDO. Users replying to *Should animal hunting be banned?* (⋆), *Should animal testing be banned?* (•) or *Should humans stop eating animals and become vegetarians?* (+) are marked in blue (pro) or red (con). We see that camps are embedded consistently across similar issues. (Color figure online)

Frames and Values. Beyond the similarity of the content of arguments, we may be interested in more abstract relations between issues – for example, we may want to investigate issues with similar frame and value distributions. To detect such issues, we compute the Frobenius norm of the difference between frame-value matrices (cf. Fig. 3) of different issues. A small Frobenius norm indicates a similar distribution of emphasized frames and values between the issues. For animal hunting, the five most similar issues revolve around animals: "*Should the United States ban the slaughter of horses for meat?*", "*Should humans stop eating animals and become vegetarians?*", "*Should animals be kept in zoos?*", "*Should we keep animals in zoos?*" and "*Should animal testing be banned?*" The next five most similar issues are "*Should cigarette smoking be banned?*", "*Should Abortion be illegal in America?*", "*Pro-life (yes) vs. pro-choice (no)?*", "*Should abortion be illegal?*" and "*Does human life begin at conception?*". Four of them are about abortion, which shows that animal rights and abortion evoke similar frames and values (see Fig. 3d), perhaps because both issues concern individuals who are unable to defend their own rights.

In the following we take a closer look at similarities and differences between the issues "*Should animal hunting be banned?*" and "*Should animal testing be banned?*" We chose these issues, as they seem similar at first glance, but reveal intriguing differences upon closer inspection. Moreover, Fig. 4 shows they have comparable camps. As expected, they mostly highlight the same frames and values (Fig. 3c). But there are also notable differences: In *animal testing*, the *health and safety* frame is expressed more often, while *capacity and resources* and *cultural identity* frames are rare.

Arguments using a *health and safety* frame for a ban on *animal hunting* or *testing* often refer to the health and safety of animals, and to the health and safety of humans when arguing against a ban. Yet, the issues raised for the health and safety of humans are not the same in arguments against a ban: for animal hunting, a common argument is that humans need meat for nutrition, which hunting helps to ensure. For animal testing the health and safety aspect often revolves around animal tests being necessary to make medicine safe for humans. This difference has also very different implications for deliberation. Concerning animal hunting, one could argue that meat for nutrition can be provided by farmed animals, or can be substituted in vegetarian diet. Finding alternatives for animal testing is more difficult and hence, needs to be addressed differently.

Concepts. Naturally, similar issues share similar concepts, for instance, *animals* in our example, while others are more distinct, e.g., *getting food* for hunting or *scientists* for animal testing. Such differences are often issue-specific and more fine-grained than differences in frames and values, as discussed above. Hence, a deeper analysis of concepts and content can help elucidate potential differences behind shared frames and values, which can be important for deliberation.

5.3 Argument Level

So far, our analysis focused on entire debates, or even collections of debates, to analyze structural properties, such as similarities and differences among debates. Yet, PAKT also supports analysis at the level of individual arguments to enable in-depth analysis. For each argument, PAKT includes abstractions to frames, values, and concepts which is what we mostly used in our analysis so far.

Beyond this, PAKT allows us to compare and relate arguments based on their content. We can do this by estimating the similarity between arguments, using either S^{3}BERT [20] or the concept overlap as another interpretable method [21].

With the computed similarities, it is almost trivial to retrieve supporting arguments (most similar among the same stance) or counterarguments (most similar but opposing stance) [31,35]. More complex argument retrieval is also easy and efficient. For example, to answer the question "*How would someone argue who wants to make a similar argument but from the perspective of value* x *instead of value* y*?*," one can use the following query which runs in ∼5 ms:

```
MATCH (:argument {id: $query_id})-[r:SIMILARITY]-(a:argument)
WHERE x in a.value AND not y in a.value
RETURN a ORDER BY r.similarity DESC
```

6 Related Work

A number of approaches have been developed with the goal of analyzing deliberative debates.

Gold *et al.* [11] propose an interactive analytical framework that combines linguistic and visual analytics to analyze the quality of deliberative communication automatically. Deliberative quality is seen as a latent unobserved variable that manifests itself in a number of observable measures and is mainly quantified based on linguistic cues and topical structure. The degree of deliberation is measured in four dimensions: i) *Participation* considers whether proponents are treated equally, i.e., whether all stakeholders are heard; ii) *Mutual Respect* is indicated by linguistic markers and patterns of turn-taking; iii) *Argumentation and Justification* aims to ensure that arguments are properly justified and refer to agreed values and understanding of the world. This is analysed using causal connectors indicating justifications, and discourse particles signaling speaker stance/attitude; iv) *Persuasiveness* measures deliberative intentions of stakeholders via types of speech acts. While Gold *et al.* focus on quality criteria that are linguistically externalized considering single arguments, our framework is targeted at revealing structural patterns in the way certain groups argue.

Bergmann *et al.* [3] are concerned with providing comprehensive overviews of ongoing debates, to make human decision makers aware of arguments and opinions related to specific topics. Their approach relies on a case-based reasoning (CBR) system that allows them to compute similarity between arguments in order to retrieve or cluster similar arguments. CBR also supports the synthesis of new arguments by extrapolating and combining existing arguments. Unlike Bergmann *et al.* who focus on grouping or retrieving related arguments, we propose a data model that focuses less on the analysis and retrieval of single arguments, but aims to provide an aggregate analysis of debates in view of their deliberative quality aspects.

Bögel *et al.* [4] have proposed a rule-based processing framework for analyzing argumentation strategies that relies on deep linguistic analysis. Their focus is on the operationalizaton of argument quality that relies on two central linguistic features: causal discourse connectives and modal particles. The proposed visualization allows users to zoom into the discourse. However, no aggregate analyses at the level of the whole debate is proposed, as we do in our paper.

Reed *et al.* have developed several tools to support the exploration and querying of arguments. ACH-Nav [37], for instance, is a tool for navigating hypotheses that offers access to contradicting hypotheses/arguments for a given hypothesis. Polemicist [18] allows users to explore people's opinions and contributions to the BBC Radio 4 Moral Maze program. ADD-up [23] is an analytical framework that analyzes online debates incrementally, allowing users to follow debates in real time. However, none of these tools are based on a data model that captures the perspectives of different stakeholders in a debate at a structural level.

VisArgue is an analytical framework by Gold *et al.* [12] that focuses on the analysis of debates on a linguistic level, focusing on discourse connectives. A novel glyph-based visualization is described that is used to represent instances where similar traits are found among different elements in the dataset. More recently, this approach has been extended to analytics of multi-party discourse [9]. The underlying system combines discourse features derived from shallow text mining with more in-depth, linguistically-motivated annotations from a discourse processing pipeline. Rather than revealing structural patterns in the way different stakeholders argument, the visualisation is designed to give a high-level overview of the content of the transcripts, based on the concept of lexical chaining.

7 Conclusion

PAKT, the Perspectivized Argumentation Knowledge Graph and Tool, introduces a pioneering framework for analyzing debates structurally and revealing patterns in argumentation across diverse stakeholders. It employs premises, conclusions, frames, and values to illuminate perspectives, while also enabling the categorization of individuals into socio-demographic groups.

Our application of PAKT to `debate.org` underscores its efficacy in conducting global analyses and offering valuable insights into argumentative perspectives. In our case studies we demonstrated the versatility of combining perspectivizing categories (*frames, values*) emphasized by different camps, in combination with concept-level analysis – which enable identification of differences within overall similarities, at the level of individual and across different issues, and how such analyses may indicate starting points for deliberation processes.

PAKT offers broad potential applications by automatically detecting imbalances or underrepresentations in arguments or debates through analyzing frames, values and concepts. Navigation through the PAKT graph via central concepts or argument-similarity edges enhances argument mining to a comprehensive level. This accessible tool allows researchers without a computer science background to explore opinion landscapes at both debate and single-argument levels. Its extensive applications include informing policy-making by dissecting contentious issues and fostering constructive discussions. Integrating PAKT into social media platforms holds promise for highlighting common ground and areas of disagreement among participants, as well as aiding moderators in identifying potentially radical or offensive content. Thus, PAKT serves as a tool to enhance understanding, and also to improve deliberative debates for all.

Limitations

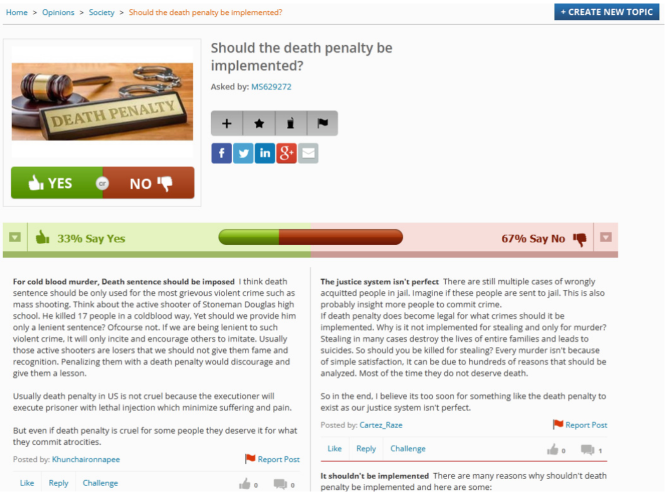
Our analysis and case study rely on automatically annotated data encompassing frames, values, and concepts. Consequently, we anticipate some degree of noise in our dataset, potentially compromising the depth of our analysis. To address this concern, we employ established methodologies derived from prior research to mitigate such discrepancies. Additionally, we perform manual annotations to gauge the quality of our data.

Our focus lies on the unique aspect of perspectivization, which is not largely explored in prior work. Consequently, we could not directly compare PAKT with other analysis tools from related studies. We hope that our discussion sparks further research, and that PAKT can serve as a valuable baseline in future work.

Lastly, our analysis and case study shed light on the practical application of PAKT in illuminating insights within debates, thereby aiding in opinion formation and decision-making processes. However, demonstrating PAKT's utility for other tasks such as moderation remains an avenue for future exploration.

Acknowledgements. We thank the anonymous reviewers for their valuable comments, and our annotators, Janosch Gehring and Rahki Asokkumar Subjagouri Nair, for their support. We also thank Marcel Nieveler for his support in the development of the web interface to PAKT. This work has been funded by the DFG through the project ACCEPT as part of the Priority Program "Robust Argumentation Machines" (SPP1999).

Appendix


Home > Opinions > Society > Should the death penalty be implemented?
+ CREATE NEW TOPIC
DEATH PENALTY
YES
or
NO
33% Say Yes
67% Say No

Should the death penalty be implemented?

Asked by: MS629272

For cold blood murder, Death sentence should be imposed I think death sentence should be only used for the most grievous violent crime such as mass shooting. Think about the active shooter of Stoneman Douglas high school. He killed 17 people in a coldblood way, Yet should we provide him only a lenient sentence? Ofcourse not. If we are being lenient to such violent crime, It will only incite and encourage others to imitate. Usually those active shooters are losers that we should not give them fame and recognition. Penalizing them with a death penalty would discourage and give them a lesson.

Usually death penalty in US is not cruel because the executioner will execute prisoner with lethal injection which minimize suffering and pain.

But even if death penalty is cruel for some people they deserve it for what they commit atrocities.

Posted by: Khunchaironnapee Report Post

Like Reply Challenge 0 0

The justice system isn't perfect There are still multiple cases of wrongly acquitted people in jail. Imagine if these people are sent to jail. This is also probably insight more people to commit crime.
If death penalty does become legal for what crimes should it be implemented. Why is it not implemented for stealing and only for murder? Stealing in many cases destroy the lives of entire families and leads to suicides. So should you be killed for stealing? Every murder isn't because of simple satisfaction, It can be due to hundreds of reasons that should be analyzed. Most of the time they do not deserve death.

So in the end, I believe its too soon for something like the death penalty to exist as our justice system isn't perfect.

Posted by: Cartez_Raze Report Post

Like Reply Challenge 0 1

It shouldn't be implemented There are many reasons why shouldn't death penalty be implemented and here are some:

Fig. 5. Screenshot of an opinion poll on `debate.org`

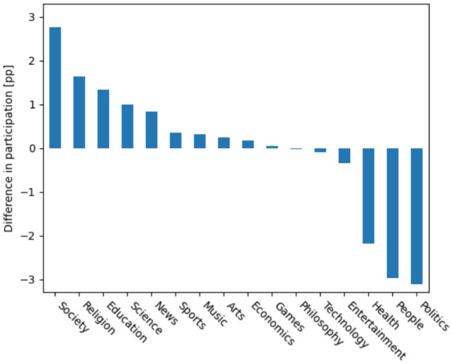
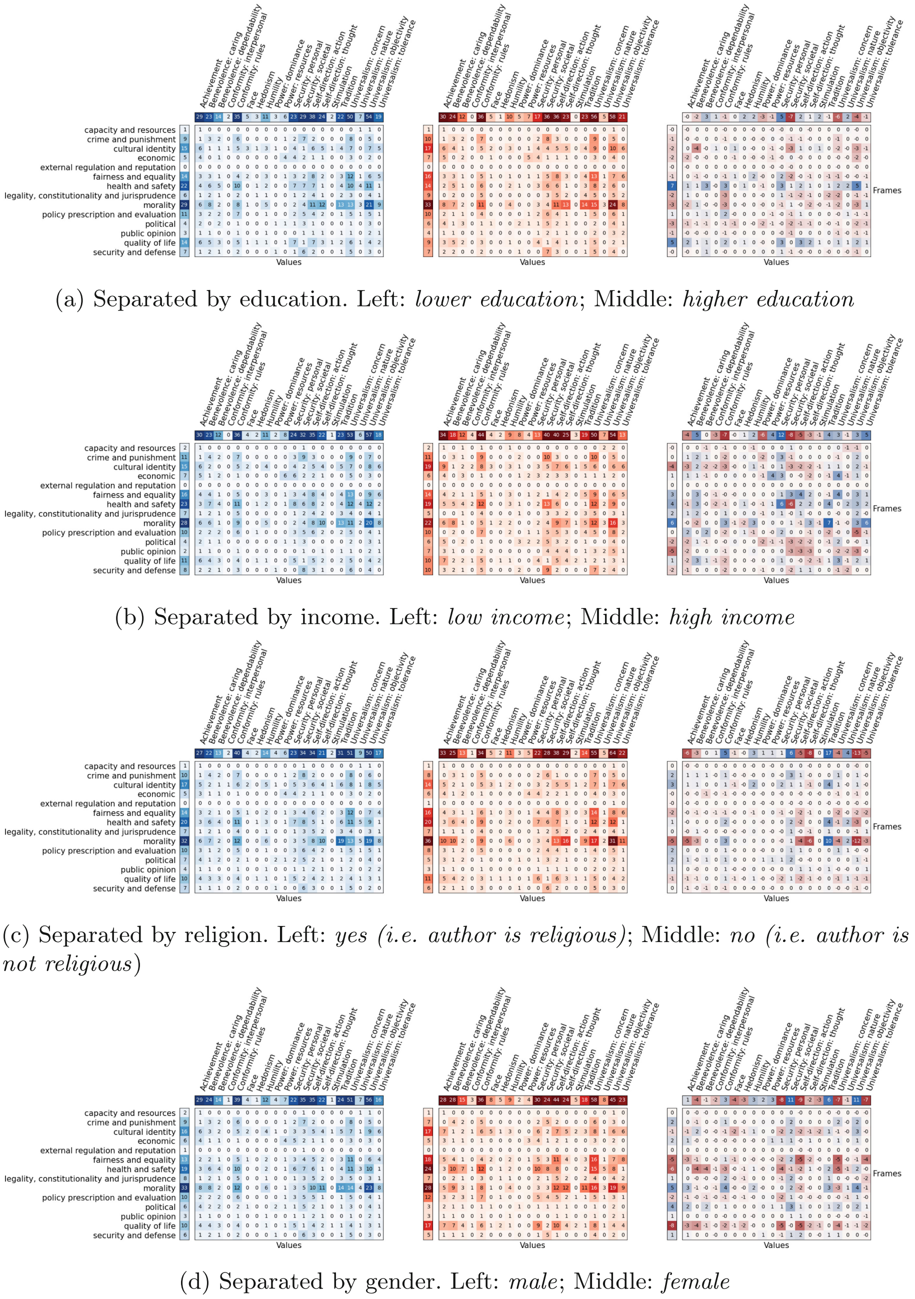


Fig. 6. Difference in relative participation between left and right winged authors.



(a) Separated by education. Left: *lower education*; Middle: *higher education*

(b) Separated by income. Left: *low income*; Middle: *high income*

(c) Separated by religion. Left: *yes (i.e. author is religious)*; Middle: *no (i.e. author is not religious)*

(d) Separated by gender. Left: *male*; Middle: *female*

Fig. 7. Comparison between frame and value matrices. The left and middle plots show distributions in percent, and the right plots show their differences in percentage points (pp). All subfigures are aggregated across all issues.

References

1. Ajjour, Y., Alshomary, M., Wachsmuth, H., Stein, B.: Modeling frames in argumentation. In: Proceedings of EMNLP-IJCNLP, pp. 2922–2932 (2019)
2. Alshomary, M., El Baff, R., Gurcke, T., Wachsmuth, H.: The moral debater: a study on the computational generation of morally framed arguments. In: Proceedings of ACL, pp. 8782–8797 (2022)
3. Bergmann, R., Schenkel, R., Dumani, L., Ollinger, S.: Recap - information retrieval and case-based reasoning for robust deliberation and synthesis of arguments in the political discourse. In: Proceedings of LWDA, pp. 49–60 (2018)
4. Bögel, T., et al.: Towards visualizing linguistic patterns of deliberation: a case study of the S21 arbitration. In: DH (2014)
5. Boydstun, A.E., Card, D., Gross, J., Resnick, P., Smith, N.A.: Tracking the Development of Media Frames within and across Policy Issues (2014)
6. Card, D., Boydstun, A.E., Gross, J.H., Resnik, P., Smith, N.A.: The media frames corpus: annotations of frames across issues. In: ACL/IJCNLP, pp. 438–444 (2015)
7. Durmus, E., Cardie, C.: Exploring the role of prior beliefs for argument persuasion. In: Proceedings of NAACL, pp. 1035–1045 (2018)
8. Durmus, E., Cardie, C.: A corpus for modeling user and language effects in argumentation on online debating. In: Proceedings of ACL, pp. 602–607 (2019)
9. El-Assady, M., Hautli-Janisz, A., Gold, V., Butt, M., Holzinger, K., Keim, D.: Interactive visual analysis of transcribed multi-party discourse. In: Proceedings of ACL, System Demonstrations, pp. 49–54 (2017)
10. Falk, N., Jundi, I., Vecchi, E.M., Lapesa, G.: Predicting moderation of deliberative arguments: is argument quality the key? In: Argument Mining, pp. 133–141 (2021)
11. Gold, V., et al.: Visual linguistic analysis of political discussions: Measuring deliberative quality. Digit. Scholarsh. Humanit. **32**(1), 141–158 (2017)
12. Gold, V., Hautli-Janisz, A., Holzinger, K., El-Assady, M.: Visargue: analysis and visualization of deliberative political communication. Polit. Commun. Rep. **26**(1), 1–2 (2016)
13. Heinisch, P., Cimiano, P.: A multi-task approach to argument frame classification at variable granularity levels. IT Inf. Technol. **63**(1), 59–72 (2021)
14. Heinisch, P., Plenz, M., Frank, A., Cimiano, P.: ACCEPT at SemEval-2023 task 3: an ensemble-based approach to multilingual framing detection. In: Proceedings of SemEval, pp. 1347–1358 (2023)
15. Kiesel, J., Alshomary, M., Handke, N., Cai, X., Wachsmuth, H., Stein, B.: Identifying the human values behind arguments. In: Proceedings of ACL, pp. 4459–4471 (2022)
16. Kiesel, J., et al.: SemEval-2023 task 4: ValueEval: identification of human values behind arguments. In: Proceedings of SemEval, pp. 2287–2303 (2023)
17. Kobbe, J., Rehbein, I., Hulpuş, I., Stuckenschmidt, H.: Exploring morality in argumentation. In: Proceedings of Argument Mining, pp. 30–40 (2020)
18. Lawrence, J., Visser, J., Reed, C.: Polemicist: a dialogical interface for exploring complex debates. In: Proceedings of COMMA, Cardiff, Wales, UK, pp. 365–366 (2022)
19. Liu, Y., et al.: Roberta: a robustly optimized bert pretraining approach (2019). arXiv
20. Opitz, J., Frank, A.: SBERT studies meaning representations: decomposing sentence embeddings into explainable semantic features. In: AACL-IJCNLP (2022)

21. Plenz, M., Heinisch, P., Frank, A., Cimiano, P.: PAKT: Perspectivized Argumentation Knowledge Graph and Tool for Deliberation Analysis (with Supplementary Materials). arXiv (2024)
22. Plenz, M., Opitz, J., Heinisch, P., Cimiano, P., Frank, A.: Similarity-weighted construction of contextualized commonsense knowledge graphs for knowledge-intense argumentation tasks. In: Proceedings of ACL, pp. 6130–6158 (2023)
23. Plüss, B., et al.: Add-up: visual analytics for augmented deliberative democracy. In: Proceedings of Computational Models of Argument, vol. 305, pp. 471–472 (2018)
24. Reiber, L.: Opening the black box of deliberation: what are arguments (really) based on? A theory-driven and exploratory analysis of the role of knowledge in the process of deliberation. Soziologiemagazin **5**(2), 149–158 (2019)
25. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using Siamese BERT-networks. In: Proceedings of EMNLP-IJCNLP, pp. 3982–3992 (2019)
26. Reimers, N., Schiller, B., Beck, T., Daxenberger, J., Stab, C., Gurevych, I.: Classification and clustering of arguments with contextualized word embeddings. In: Proceedings of ACL, pp. 567–578 (2019)
27. Ruckdeschel, M., Wiedemann, G.: Boundary detection and categorization of argument aspects via supervised learning. In: Argument Mining, pp. 126–136 (2022)
28. Schiller, B., Daxenberger, J., Gurevych, I.: Aspect-controlled neural argument generation. In: Proceedings of NAACL-HLT, pp. 380–396 (2021)
29. Schroter, D., Dementieva, D., Groh, G.: Adam-smith at SemEval-2023 task 4: discovering human values in arguments with ensembles of transformer-based models. In: Proceedings of SemEval, pp. 532–541 (2023)
30. Schwartz, S.H.: Are there universal aspects in the structure and contents of human values? J. Soc. Issues **50**(4), 19–45 (1994)
31. Shi, H., Cao, S., Nguyen, C.T.: Revisiting the role of similarity and dissimilarity in best counter argument retrieval. ArXiv (2023)
32. Speer, R., Chin, J., Havasi, C.: Conceptnet 5.5: an open multilingual graph of general knowledge. In: Proceedings of AAAI, pp. 4444–4451 (2017)
33. Stab, C., Gurevych, I.: Parsing argumentation structures in persuasive essays. Comput. Linguist. **43**(3), 619–659 (2017)
34. Wachsmuth, H., et al.: Building an argument search engine for the web. In: Proceedings of Argument Mining, pp. 49–59 (2017)
35. Wachsmuth, H., Syed, S., Stein, B.: Retrieval of the best counterargument without prior topic knowledge. In: Proceedings of ACL, pp. 241–251 (2018)
36. Xie, J.Y., Ferreira Pinto Junior, R., Hirst, G., Xu, Y.: Text-based inference of moral sentiment change. In: Proceedings of EMNLP-IJCNLP, pp. 4654–4663 (2019)
37. Zografistou, D., Visser, J., Lawrence, J., Reed, C.: ACH-Nav: argument navigation using techniques for intelligence analysis. In: Proceedings of COMMA, pp. 377–378 (2022)

PolArg: Unsupervised Polarity Prediction of Arguments in Real-Time Online Conversations

Mirko Lenz[1](✉) and Ralph Bergmann[1,2]

[1] Trier University, Universitätsring 15, 54296 Trier, Germany
info@mirko-lenz.de, bergmann@uni-trier.de
[2] Branch Trier University, German Research Center for Artificial Intelligence (DFKI), Behringstr. 21, 54296 Trier, Germany
ralph.bergmann@dfki.de

Abstract. The increasing usage of social networks has led to a growing number of discussions on the Internet that are a valuable source of argumentation that occurs in real time. Such conversations are often made up of a large number of participants and are characterized by a fast pace. Platforms like X/Twitter and Hacker News (HN) allow users to respond to other users' posts, leading to a tree-like structure. Previous work focused on training supervised models on datasets obtained from debate portals like Kialo where authors provide polarity labels (i.e., support/attack) together with their posts. Such classifiers may yield suboptimal predictions for the noisier posts from X or HN, so we propose unsupervised prompting strategies for large language models instead. Our experimental evaluation found this approach to be more effective for X conversations than a model fine-tuned on Kialo debates, but less effective for HN posts (which are more technical and less argumentative). Finally, we provide an open-source application for converting discussions on these platforms into argument graphs.

Keywords: Argumentation · Argument Graphs · Argument Mining · Large Language Models · Social Networks · Datasets · Open Source

1 Introduction

Argumentation is a fundamental part of human communication and can be found in many different forms. Having the best argument in a conversation is often a key factor to success. Computational Argumentation (CA) consequently has the potential of supporting a wide range of user types—ranging from journalists researching for their next article to students writing their thesis. Most arguments are expressed in natural language, which means that machines first need to parse the argumentative structures within a text through a process called Argument Mining (AM) [18]. With the advent of the Internet, there is a growing number of discussions happening on platforms such as X/Twitter, Reddit, and

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 108–126, 2024.
https://doi.org/10.1007/978-3-031-63536-6_7

Hacker News (HN). These new forms of discourse are characterized by a large number of participants and a fast pace and share one common trait: Users can respond directly to other users' posts, leading to a tree-like structure of the conversation. Compared to plain texts, this allows users to focus on certain parts of the discussion more easily—for instance, by hiding certain parts of the tree.

Discussions on social networks often involve argumentation (e.g., if users try to convince others of their opinion) [14], thus we argue that these platforms are a valuable resource for CA. Imagine an emerging event, such as the release of a new product or a political scandal. In such a situation, it is important to be able to quickly identify the most important arguments—both for experts like journalists and laymen. Curated argumentation databases cannot be used for evolving topics, so this is mostly a manual process at the moment. With the prestructured conversations from social networks, only two tasks are left to use them as argument graphs: (i) Identifying which of the posts are actually argumentative and (ii) determining the polarity (support or attack) between them. Both have already been tackled in previous work (see Sect. 3), but existing approaches rely only on supervised classifiers. This means that they need a large amount of *annotated* data to train on, which is not available for social networks like X. Instead, most datasets are obtained from moderated debate portals like Kialo. Contrary to most social networks, posts on these platforms are moderated and users tend to write elaborate replies. The polarity of the replies is explicitly stated, making it relatively easy to train supervised models. We found that the resulting models are not directly applicable to other types of data (e.g., social network posts), requiring the creation of training data from scratch.

To remove the need to annotate social network posts, we propose an unsupervised approach based on prompting strategies for Large Language Models (LLMs). In our paper, we focus on the polarity prediction task, leaving the identification of argumentative posts to future work (see Sect. 7). Consequently, we pursue the following research question: *Can unsupervised LLMs match or even surpass the polarity prediction quality of supervised approaches?* Our main contributions are (i) Four different prompting strategies for different types of LLMs to predict the polarity between two posts in a conversation, (ii) an extensive experimental evaluation on an existing benchmark corpus as well as two new datasets obtained from the platforms X and HN, and (iii) an open-source application that allows users to perform real-time AM on these two platforms.

The remainder of this paper is structured as follows: Sect. 2 introduces fundamental concepts, followed by a review of related work in Sect. 3. Section 4 presents the prompting strategies that are evaluated in Sect. 5. We discuss limitations in Sect. 6 and conclude our paper in Sect. 7.

2 Foundations

In the following section, we introduce the most important concepts of CA and Natural Language Processing (NLP) [2] as well as the conversation platforms used in this paper.

2.1 Computational Argumentation

Before dealing with CA, we start with the concept of an *argument*, often defined as a single *claim* and several supporting or attacking *premises* [21,24]. Both claims and premises are the fundamental elements of argumentation, also known as Argumentative Discourse Units (ADUs) [21], and can range from a few words to complete paragraphs. Most argumentative texts revolve around a primary claim that the author aims to establish, known as the *major claim* [25].

A graph-based format is an intuitive way to represent these structures, leading to the concept of *argument graphs*. In our paper, we use an extended version of the Argument Interchange Format (AIF) [9] and consider it as a triple $G = (V, E, M)$, where all ADUs are nodes or vertices V, the relationships between them form the edges $E \subseteq V \times V$, and M representing its major claim. The graph includes *atom nodes* A representing individual ADUs and *scheme nodes* S denoting the type of connection (i.e., support/attack) between other nodes. Thus, the set of nodes V can be expressed as $V = A \cup S$. In this structure, edges are not allowed to connect two atom nodes by definition, so the set of edges E can be defined as $E = V \times V \setminus A \times A$.

The term AM refers to the process of extracting and identifying argumentative structures from textual data—for instance, detection claims and premises and predicting relations between them. Our work contributes to the latter task, which is also known as *polarity prediction*: "Does a premise *support* or *attack* the claim?" Cayrol and Lagasquie-Schiex [7] introduced a *Bipolar Argumentation Framework* to represent these relations. We stick to the AIF standard and its scheme nodes introduced earlier, so we refer the interested reader to their work for a more formal definition of this framework. By combining multiple AM tasks, complex argument graphs can be constructed.

2.2 Natural Language Processing

The field of NLP offers a wide range of techniques to process natural language texts. When dealing with structured argumentation in the form of graphs, the aforementioned atom nodes contain texts that can be processed through NLP. Since the inception of representing words through embeddings, the concept has evolved to transformer-based models popularized by Bidirectional Encoder Representations from Transformers (BERT) [11]. These models are pre-trained on a large corpus of texts and can then be fine-tuned on a specific task—for instance, predicting Textual Entailment (TE) [16]. TE—also known as Natural Language Inference (NLI)—is the task of determining whether a given text *entails* another text and is conceptually similar to the investigated polarity prediction task. However, datasets for TE are not directly applicable to polarity prediction, since the notion of *entailment/contradiction* is not the same as *support/attack*: For example, a premise may *entail* a claim, but does not necessarily *support* it.

Based on the transformer architecture, LLMs having billions of parameters have been developed in recent years. In addition to fine-tuning, they can be used in a chat-based way by *prompting* them for some output. This approach is also

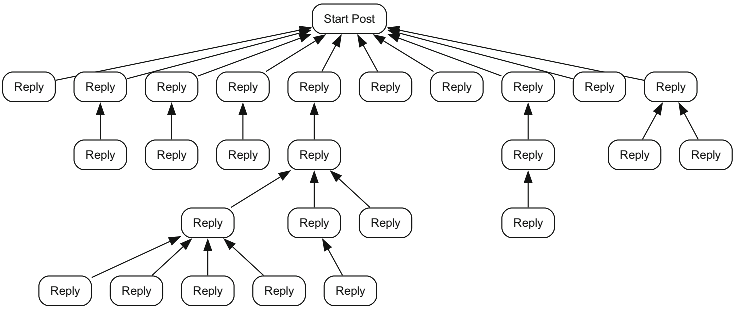


Fig. 1. Fragment of a conversation from the platform Hacker News with the text of the posts replaced by their type. (Source: https://news.ycombinator.com/item?id=37744339)

known as *few-shot learning* [27] since the model does not need to be trained on a large dataset. LLMs differ w.r.t. their maximum context length—that is, the number of tokens they can process at once. As we will discuss in Sect. 4, this is an important factor to consider when designing prompts.

2.3 Online Conversation Platforms

Having introduced argumentation, the use of graphs in this context, and the most important concepts of LLMs, we now detail the unique characteristics of the different conversation platforms with which we are concerned in this paper: Kialo, X, and HN. The first is a moderated debate portal, whereas the other two are social networks. One common feature is that users can respond to the posts of other users, leading to a tree-like structure of the conversation like the one shown in Fig. 1. These trees depict a special type of argument graph, where each scheme node has exactly one incoming and one outgoing connection to some atom node. The starting post of the conversation can be seen as the major claim of the argument graph. Therefore, we can specialize the definition of an argument graph $G = (V, E, M)$ by redefining the set of edges $E = A \times S \cup S \times A$ and setting constraints for the number of outgoing and incoming edges for scheme nodes $\forall s \in S : \text{outdegree}(s) = \text{indegree}(s) = 1$.

A central difference between the three platforms is the type of posts they contain: Kialo[1] is a platform that aims to facilitate high-quality debates by providing a structured environment for users to discuss a wide range of topics. Users not only reply to another user's post, but also explicitly state the polarity of their reply. X[2] (formerly known as *Twitter*) is a social network where users can post short messages (formerly known as *tweets*) that are limited to 280 characters. Similar to Kialo, other users can reply to these tweets, but the polarity or even the stance of their post is unknown. At the same time, X has additional features

[1] https://www.kialo.com/.
[2] https://x.com and https://twitter.com.

like *quotes*, *mentions*, and *hashtags* that can be used to refer to other posts. For example, a reply on X may contain multiple mentions of other users, leading to a more complex structure than the hierarchical conversations found on Kialo and HN. In our paper, we focus on the tree-like reply structure and leave the remaining features for future work. HN[3] is a social news website run by the venture capital firm *Y-Combinator* where users can submit links to articles or ask questions and other users can comment on them. The platform is primarily aimed at developers, and discussions are often more technical in nature.

3 Related Work

In the following section, we highlight some of the most important contributions to the field of AM and CA concerned with online conversations. This field of research has received a steady stream of contributions in the last decade, of which we selected the works that are most relevant to our paper. For readers more interested in text mining approaches for tweets that have been proposed in that timeframe, we refer to the study conducted by Karami et al. [15].

The baseline model used in Sect. 5 is based on the work of Agarwal et al. [1]. The core of their contribution is a deep learning architecture dubbed GraphNLI that predicts the polarity between two posts in a threaded conversation. Instead of relying solely on the textual content of two posts, *graph walks* are used to sample contextual information from nearby nodes in the thread and thus generate richer embeddings. The authors evaluated their approach on debates obtained from the Kialo platform and compared it to four baseline approaches—one of them being a Sentence-BERT (S-BERT) [22] based classifier that only receives the two posts without any context. The results showed that GraphNLI outperformed all baselines on the polarity prediction task, although the difference to the S-BERT classifier in terms of accuracy/precision/recall was rather small (approximately 3%). In an ablation study, the ancestor nodes were found to be relevant to the context than the child nodes. With the graph walks being based on probabilities assigned to nodes, the results are not deterministic. Evaluation of GraphNLI on Twitter data is left for future work.

Other datasets containing argumentative conversations have been proposed in the past—for instance, based on the *Debatepedia* website [5,6]. Bosc et al. [3] created the DART dataset that contains tweets (among other topics) related to the Apple Watch release. At that time, it was not possible to fetch the entire conversation tree, so the authors resorted to heuristics to predict pairs of tweets—meaning that the original structure of the conversation is lost. There also exists a large body of work on AM for social media conversations, ranging from the identification of ADUs [13] to the detection of opinions given some tweet [13,20]. The mentioned DART dataset has been used to identify argumentative tweets and predict their polarity [4] and to recognize facts and sources in tweets [12].

[3] https://news.ycombinator.com/.

Table 1. Matrix with characteristics of our prompting strategies.

	Isolated	**Sequential**	**Contextualized**	**Batched**
Includes context	×	✓	✓	✓
Parallel predictions	✓	×	✓	✓
Usable without JSON schemas	✓	✓	✓	×
Required context length	small	medium	small	large
Number of predictions for n pairs	n	n	n	1

4 Prompting Strategies

As mentioned in Sect. 2.2, the use of LLMs shifts the focus from feature engineering and model design to the so-called *prompt engineering*. In the following section, we present four different strategies for predicting the polarity between two posts in a conversation. All of them are *zero-shot* approaches—that is, no exemplary cases are given to the model—since we aim at providing a universally applicable solution for different kinds of conversation. Each strategy is tailored for a different kind of LLM, depending on its capabilities. To estimate the required context size of a model, we distinguish between categories *small* (100–500 tokens), *medium* (500–5,000 tokens), and *large* (more than 5,000 tokens). One strategy makes use of JSON-based responses enforcing a given schema through OpenAI's *function calling* feature, rendering it unusable for other models.

The main difference between the strategies is the amount of context they provide to the model. While the first two strategies only use the tree structure to identify premises and their claims—making them applicable to any kind of conversation—the latter two use it to provide additional information to the model: The *isolated* strategy does not use any context at all, while the *sequential* one provides the model with all previous requests and responses. The *contextualized* strategy samples nearby nodes in the conversation tree, and the *batched* one passes the entire conversation as context. They are designed to work equally well for smaller conversations that have only a few posts, as well as for larger ones that potentially contain hundreds of posts. As part of our evaluation in Sect. 5, a diverse set of graph sizes is used to verify this. A comparison matrix can be found in Table 1 and concrete prompts in Appendix A.

4.1 Isolated Prompting

In this rather intuitive approach, we simply feed two posts into the model without any additional context from the conversation—that is, we assume that they are self-contained. As part of the system message, the model is instructed to predict the polarity between a premise and its claim and respond with "support" or "attack". This approach can be applied to virtually any LLM and is therefore a good starting point for our evaluation. Since all predictions are separate from each other, one can query the model for all of them in parallel—making inference faster for multi-GPU setups. We observed that LLMs may produce more text than the single word it is supposed to output, which we deal with by performing substring matching. For example, if the model outputs "I support this claim.", we would consider this as a prediction of "support".

4.2 Sequential Prompting

The basic idea is the same as in the previous approach, but we simulate memory by storing all previous predictions for a single post and passing this conversation history to the model. This could make it possible for the model to provide predictions that are consistent with previous decisions, and thus potentially increase the accuracy. The first prediction for a post still does not have any context—the difference only becomes apparent from the second prediction onward. Since the number of messages increases linearly with the number of posts in a conversation, this strategy is not suitable for LLMs with a limited context size. One can remedy this by removing some of the earlier posts and their predictions from the history, but this would also remove the context for the corresponding posts. Compared to isolated prompting, this strategy cannot be run in parallel.

4.3 Contextualized Prompting

This strategy is an extension of the isolated and sequential prompting approaches that aims to solve their limitations. The isolated technique misses any kind of contextual information, potentially leading to wrong predictions. The sequential approach might be prone to subsequent errors: Wrong predictions for the first requests may influence the model's decision for later ones.

To solve these problems, we propose to sample nearby nodes for contextual information in a similar way to GraphNLI. Agarwal et al. [1] proposed the use of *random* graph walks (see Sect. 3), which means that the results can change between runs. The authors found that providing four nodes as a context yielded the best results, so we propose the following *deterministic* sampling technique: Choose one parent node of the claim, one child node of the premise, and one sibling node of each (if available), resulting in a maximum of four nearby nodes. In case there are multiple candidates, choose the one with the longest text—this should provide the model with the most information available in nearby nodes. A consequence of this sampling is that some nodes in the graph may have limited context—most notably leaf nodes without siblings—even in large discussions.

While in theory this approach could be applied to both the isolated and sequential prompting, we only use it for the former since the latter already includes context, and we did not find any benefit in combining both techniques. Contextualized prompting will naturally need a larger context length than the isolated approach, but the token size does not scale linearly with the number of posts—consequently, it may be used with LLMs having limited context sizes.

4.4 Batched Prompting

All previous approaches fed the argument pairs to the model one by one, but with the development of LLMs having context sizes of more than 100,000 tokens, we gain the option of passing all pairs in a single request. It would still be possible to perform a single prediction, but that would be inefficient. Instead, this strategy uses another feature that some LLMs (e.g., those created by OpenAI): The ability to handle structured JavaScript Object Notation (JSON) data—also known as *function calling*. This enables us to use a single request to predict the polarity between all pairs in a conversation. We expect this strategy to show the best efficiency since the model can use the entire conversation as a context.

When querying a LLM with such a complex request, there is a chance of hallucinations—for instance, the model might predict a polarity between two posts that are not connected in the conversation or even come up with posts on its own that are not part of the conversation. In an effort to mitigate these, we append a unique identifier to each premise and claim and use only those predictions that match the corresponding identifiers. In case some predictions are missing, we perform a second request for the missing ones only and provide the available predictions as a context.

5 Experimental Evaluation

In the next section, we present the datasets used for our evaluation, followed by the experimental setup. We then proceed with the results of our experiments and discuss their implications. We start by introducing our hypotheses to answer our research question formulated in Sect. 1: *Can unsupervised LLMs match or even surpass the polarity prediction quality of supervised approaches?*

H1. The prediction quality of supervised models is influenced by the type of posts in the training data (i.e., debate portals vs. social networks).

H2. Adding context information to the prompts improves the prediction quality of the model.

H3. At least one of our prompting strategies matches or exceeds the prediction quality of established supervised approaches.

H1 aims at showcasing the difficulties in transferring models between different types of posts, whereas H3 checks that our prompting strategies are also applicable to high-quality debates. H2 test which of strategies presented in Sect. 4 performs best on different types of data.

5.1 Experimental Setup

In order to assess our hypotheses, we implemented our approach in Python and made the source code publicly available on GitHub under the permissive MIT license.[4] Our application is implemented through a client-server architecture, which means that other developers can easily integrate it into their own projects. To demonstrate this, we built another open-source application called xArgueBuf that enables real-time AM on X and HN.[5] Throughout this evaluation, we use a set of standard classification metrics, namely *accuracy* A, *precision* P, and *recall* R. Furthermore, we tested the statistical significance of our results using *McNemar's test* [19] (χ^2 distribution, continuity correction, significance level $\alpha = 0.01$). The test is based on disconcordant pairs in a contingency table and allows us to assess the difference in prediction quality between two approaches when using the same data. Its null hypothesis states that the two approaches are equally good at predicting the polarity between posts.

As LLMs for our evaluation, we use the proprietary ChatGPT developed by OpenAI and the open Llama 2 [26] developed by Facebook. The prompting strategies that require small to medium context length were tested on the `gpt-3.5-turbo-1106` model, whereas the batched one requiring a larger context size was tested on the `gpt-4-1106-preview` model (also known as GPT-4 Turbo). The tests involving Llama all use the model with 13 billion parameters fine-tuned on the chat task. Language models tend to provide unpredictable output, so for each prompt-based evaluation, we provide the number of missing predictions (N/A) as a percentage. Due to load-balancing measures implemented by OpenAI, we could not utilize the full context length of their largest model in a deterministic manner—some requests would randomly time out. For the batched strategy, we thus limited one request to 50 claim-premise pairs and performed multiple requests if necessary.

To compare our prompting strategies with established supervised approaches, we used the same baseline model as Agarwal et al. [1]: A cross-encoder based on the S-BERT architecture.[6] Compared to a regular bi-encoder where the two posts are encoded separately, both posts are passed simultaneously to the transformer. Agarwal et al. [1] report results that almost match their GraphNLI model, so we expect this baseline to be a good indicator for the effectiveness of our prompting strategies. We trained multiple variants of this baseline model on different datasets (see next section) to test H1.

5.2 Datasets

In the following section, we present the three datasets used in our evaluation: Two new ones containing conversations from X and HN as well as the dataset used by Agarwal et al. [1] to evaluate their GraphNLI model. One goal of our paper is to facilitate real-time argument mining, so our methods should be applicable to conversations of different sizes and shapes, including small ones containing only a few posts. An overview of the number of posts contained in them is shown in Fig. 2, showing that a wide range of conversation sizes is covered. The part

[4] https://github.com/recap-utr/polarg.

[5] https://github.com/recap-utr/xarguebuf.

[6] The same pre-trained model (`distilroberta-base`) is used.

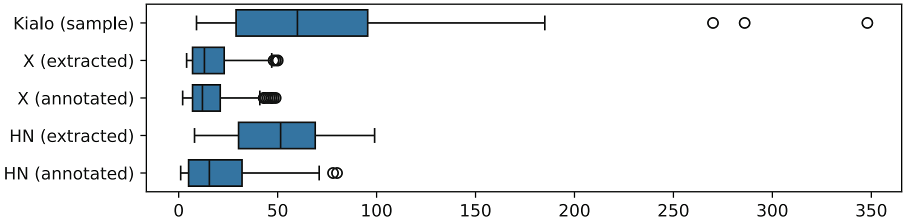


Fig. 2. Distribution of the number of posts in the datasets used in our evaluation.

of the GraphNLI dataset used in our evaluation is rather large—conversations on average consist of 86 posts, some even having more than 200 posts—whereas the newly annotated X and HN datasets on average contain 15 and 21 posts, respectively. Although the X and HN datasets are static snapshots, their diverse sizes and shapes should therefore approximately resemble the conversations that would be encountered in a real-time scenario.

The GraphNLI corpus [1] has been crawled from Kialo and contains a total of 1,560 conversations with 327,579 posts. Since these debates already include polarity labels, manual annotation was not necessary. They also did not need to remove non-argumentative posts from the conversations due given that Kialo is a moderated platform focused on high-quality discourses. Due to the rather large size of the dataset and the rate limits imposed by OpenAI (see above), we sampled 10% of the debates from our test dataset to be used for our evaluation. This test ultimately contains 31 graphs.

The other two datasets containing posts from X and HN have been created from scratch for this paper, as we are not aware of any existing ones that are suitable for our evaluation. After downloading the conversations via the platform's Application Programming Interface (API), the conversation trees were then handed over to two student experts who removed posts that did not contain argumentative content and assigned a polarity (i.e., support/attack) to each missing scheme node. These new corpora are available on request from the authors to other researchers for non-commercial purposes. In the following, we briefly discuss the queries used to obtain the data, the difficulties we faced during the annotation process, and the reliability of the resulting datasets.

To train our baseline classification model, each dataset has been divided into three parts: 80% for training and 20% for testing. The training set was further divided into 80% for training and 20% for validation. The splits were made at the conversation level to ensure that all posts of a single conversation were in the same set to avoid data leakage.

X Dataset. This corpus contains posts related to the 2020 presidential election in the United States. Our query is based on hashtags identified in previous studies [8,23]. Here is the complete list of hashtags used in our query:

#2020election, #2020elections, #4moreyears, #americafirst, #biden, #biden2020, #bidenharris2020, #bluewave2020, #covid19, #debate2020, #donaldtrump, #draintheswamp,

#election2020, #electionday, #elections_2020, #elections2020, #fourmoreyears, #gop, #joebiden, #kag, #kag2020, #keepamericagreat, #latinosfortrump, #maga, #maga2020, #makeamericagreatagain, #mypresident, #november3rd, #novemberiscoming, #patriotismwins, #qanon, #redwave, #stopthesteal, #trump, #trump2020, #trump2020landslide, #trumphasnoplan, #trumpliespeopledie, #trumppence2020, #trumpvirus, #uselections, #vote, #vote2020, #votebluetosaveamerica, #votered, #voteredlikeyourlifedependsonit, #voteredtosaveamerica, #votetrump2020, #votetrumpout, #yourchoice, #americafirst

These hashtags were joined together using the logical or operator ($\vee$). Since this query was only used to find the *starting post* of a conversation, we further restricted the set of results using the following constraints: (i) The post must be published between 3rd June 2020 (i.e., start of primaries in Iowa) and 2nd November 2020 (day before election), (ii) is must be written in English, (iii) the author must be verified by Twitter, and (iv) the post must not be a retweet, reply, or quote. When downloading the dataset on 8th December 2022, more than 2,000,000 posts matched these criteria. In other words, we identified over two million conversations, each containing possibly hundreds or even thousands of replies. X's API does not allow filtering by likes, followers, or other metrics, so we decided to let X order the posts by *relevancy* and use the best 500 posts for our annotation process. Our rationale here is that the most relevant posts for X are likely also those that appear in the *For You* tab on their website and app, so this choice should closely mimic the experience of a regular user. For each of the resulting 500 posts, we recursively fetched all replies (i.e., the entire conversation) from X's API, resulting in a file containing more than 2.5 GB of compressed textual data.

Handing over such a large amount of data to our annotators would have been impractical, so we decided to further reduce the number of tweets by applying the following constraints: (i) Each post must have at least 20 characters (otherwise it is unlikely to contain valuable and argumentative information), (ii) each post must have at least one interaction (i.e., like, retweet, reply, or quote), (iii) the depth of a conversation must be at least two (i.e., the distance between the starting post and a leaf reply must be at least two), and (iv) a conversation must afterwards have at least three and at most 50 posts left. The last constraint is necessary to ensure that the annotation process is feasible for our experts. With these restrictions, we were left with 294 conversations that contain 4,930 posts in total. During the annotation process, the experts remove all posts that are not argumentative, leading to a final size of 272 conversations with 4,067 posts. The relatively low number of posts removed during the process again shows that social networks contain a good amount of argumentative content.

Hacker News Dataset. We already stated the differences between HN and X in Sect. 2.3, but it essentially boils down to the fact that HN is a platform targeted at a more technical audience without restrictions on the number of characters. This means that we are not faced with the issue of filtering millions of posts, and thus we used a simpler method to obtain the data. Their API does not natively support arbitrary queries, so we opt to take snapshots of the *best* posts at two different points in time: On 5th October 2023 and 30th October 2023 (about two weeks apart to let enough new posts emerge). We fetched regular posts and *Ask HN* posts separately and merged them afterwards—there was only one overlapping post between both sets.

But even on HN, this approach resulted in almost 1,000 posts, so we again settled on some constraints to filter out the most promising conversations: (i) The starting post must have received a minimum number of 10 upvotes, (ii) each conversation must have at least ten and at most 100 replies, (iii) each reply must have at least 20 characters, and (iv) the depth of a conversation must be at least two (i.e., the distance between the starting post and a leaf reply must be at least two). These constraints resulted in 206 conversations with 10,596 posts in total. The conversation depicted in Fig. 1 is an example of the type of discussion we extracted from the API. After the manual annotation process, we were left with 198 conversations that contained 4,190 posts. This means that more than half of the posts were deemed not argumentative. From our experience, this seems to stem mainly from the fact that the posts on HN contain a lot of factual information instead of opinions. For example, when trying to answer a question in the format *Ask HN*, users are likely to provide a direct answer rather than argue for a certain position. Even if a reply to such a factual post might then contain some argumentative information, we remove the entire branch from the conversation tree to be consistent with the annotation process for the X dataset.

Annotation Reliability. During the initial annotation process, each annotator processed a different set of conversations, which means that no Inter-Annotator Agreement (IAA) could be calculated. We also did not have the resources to have each conversation annotated by two experts. To still ensure the reliability of the annotations, we took a random 30% sample of the unannotated X and HN datasets and handed them over to a team of three student experts—more specifically, the team that also labeled the HN dataset. We designed the sampling process in a way that no expert would annotate a conversation they had already seen before. Upon completion, a total of 8,938 scheme nodes had labels by two independent annotators for which we calculated the IAA using Cohen's κ [10].

During the annotation, the experts were free to remove non-argumentative relations, thus we consider two different perspectives: (i) The IAA for the entire dataset (including schemes removed by the annotators), and (ii) the IAA for the subset of scheme nodes that were labeled as either *support* or *attack* by both annotators (i.e., those considered to be argumentative). We received κ values of (i) .434 and (ii) .638 for the X dataset and (i) .202 and (ii) .410 for the HN dataset. Based on the Landis and Koch guidelines [17], we consider the IAA for perspective one (i.e., the entire dataset) to be *moderate* for X and *fair* for HN. For perspective two (i.e., the subset of argumentative schemes), we consider the IAA to be *substantial* for X and *moderate* for HN. The implications of these results are twofold: First, the IAA for the subset of argumentative schemes is higher than for the entire data set, meaning that labeling argumentative content was easier. Second, the IAA for the X dataset is higher than for the HN dataset, indicating X posts are more argumentative HN posts. As stated in Sect. 1, we leave the detection of argumentative content to future work.

Table 2. Effectiveness results of different variants of the baseline model with the best metrics for each test dataset marked in bold.

Test Dataset	Train Dataset	*A*	*P*	*R*
Kialo	Kialo	.752	.780	.752
Kialo	X ∪ HN	.636	.641	.573
Kialo	All	**.782**	**.785**	**.762**
HN	Kialo	.708	**.642**	.554
HN	X ∪ HN	.700	.612	.612
HN	All	**.715**	.628	**.643**
X	Kialo	.689	.552	.557
X	X ∪ HN	**.753**	**.649**	.625
X	All	.748	.628	**.671**

5.3 Results and Discussion

Having described our experimental setups and the datasets used for our evaluation, we now present our results and discuss them in detail. We start by investigating the effectiveness of our baseline model using Table 2 to answer H1. For the Kialo and HN dataset, the difference between a classifier trained on Kialo graphs and a combination of the three sites is negligible. For posts from X, however, the effectiveness of the model trained on the Kialo dataset is considerably worse than the other two: The model trained on the much smaller X and HN is even the most efficient. Another interesting observation is that this model is considerably less effective on the Kialo test set than the other two. This means that we can only partially confirm H1: Although there is an impact for using Kialo as training data for X posts (and vice versa), HN posts did not show much difference w.r.t. the training data. This seems to strengthen the assumption that the HN posts are more similar to Kialo than they are to X.

The remaining hypotheses can be tested with the results presented in Table 3, starting with the impact of adding context information to the prompts (H2). First, we check whether the contextualized prompting strategy is more effective than the isolated one. For four of our six test cases, we observe a small improvement. However, for the other two, the isolated strategy is more effective. Comparing the isolated and contextualized strategies using McNemar's test yields a p-value of 0.23 across all models and datasets, meaning that the null hypothesis cannot be rejected. Since Agarwal et al. [1] found that adding nearby nodes is beneficial, this could be a consequence of our deterministic sampling method. The results are different for the context-aware batched prompting strategy: For all test cases, we observed notable improvements across all metrics. The comparison of isolated vs. batched and contextualized vs. batched prompting using McNemar's test yields a p-value < 0.001 in both cases, meaning that the null hypothesis can be rejected. This confirms our intuition that passing the whole conversation as context to the model is indeed beneficial. Since this strategy is

Table 3. Effectiveness results of different prompting strategies with LLMs with the best metrics for each test dataset marked in bold.

Test Dataset	**Model**	**Prompting**	*A*	*P*	*R*	N/A
Kialo	ChatGPT	Isolated	.593	.564	.720	0.73%
Kialo	ChatGPT	Contextualized	.559	.533	.753	0.15%
Kialo	ChatGPT	Batched	**.840**	**.843**	.841	1.57%
Kialo	Llama	Isolated	.540	.516	**.881**	1.53%
Kialo	Llama	Contextualized	.557	.528	.864	0.04%
HN	ChatGPT	Isolated	.578	.468	.663	0.00%
HN	ChatGPT	Contextualized	.547	.447	.724	0.00%
HN	ChatGPT	Batched	**.618**	**.504**	.686	0.13%
HN	Llama	Isolated	.480	.413	**.827**	0.00%
HN	Llama	Contextualized	.503	.422	.769	0.13%
X	ChatGPT	Isolated	.656	.505	.467	0.34%
X	ChatGPT	Contextualized	.652	.500	.577	0.34%
X	ChatGPT	Batched	**.755**	**.629**	**.752**	2.03%
X	Llama	Isolated	.556	.428	.651	12.77%
X	Llama	Contextualized	.523	.403	.750	4.63%

only possible with the largest GPT model, we cannot compare it to the Llama model. Therefore, we accept H2.

Finally, we check whether our prompting strategies match the effectiveness of the baseline model (H3) by comparing the results of the supervised model trained on all three corpora to the predictions obtained using the batched strategy. For the X dataset, McNemar's test yields a p-value of 0.61 and thus shows that there is no significant difference between the two models. For Kialo and HN, the test yields a p-value < 0.001, leading to the rejection of the null hypothesis. Closer inspection of the classification metrics reveals that in case of Kialo, the batched strategy is more effective than the baseline model, while for HN, the opposite is true. Bearing its low IAA and weak overall scores in mind, this is yet another indicator of the rather technical nature of HN posts—potentially leading to a higher uncertainty in the predictions. Even when considering that ChatGPT may have been trained on some publicly available Kialo debates and may thus be biased towards them, the effectiveness on the new X dataset shows the potential of the batched strategy. Combining all findings, we tend to cautiously confirm H3—at least for clearly argumentative posts.

5.4 Qualitative Error Analysis

Besides the quantitative results, we also performed a qualitative analysis to better understand the errors made by the LLMs. The batched one is the most promising one, so we focus on it in our analysis. Please note that due to copyright issues, we cannot provide examples of actual posts, so we discuss the context and the types of errors made and provide suggestions for improvement.

For the X dataset, we observed that the model often struggles with predictions involving posts that contain insults, sarcasm, or jokes. For example, the polarity between a factual premise and an insulting claim is often predicted differently than by the human annotator. We also identified multiple cases where replies (i.e., the premises) to posts with a negative sentiment (i.e., the claims) were predicted as support by the annotator but as attack by the model. This could be caused by the model comparing the premise to major claim instead of the directly connected claim. Another common source of errors are posts that contain emojis—especially if multiple emojis are used in a single post. Although the experts were able to interpret them correctly, the LLMs may lack the necessary context to do so. One possible solution to this problem could be to encode the emojis via a textual description.

For the HN dataset, we observed the same issues with posts containing insults or negative sentiment. In addition, we found multiple instances where the prediction of the LLMs was different from the expert's opinion, but still plausible or even a better fit. For example, an expert labeled the relation between a premise supporting a claim that in turn attacks another claim as *attack*, while the model correctly predicted *support*. This again shows the inherent subjectivity of the tasks and confirms our finding that the IAA for the HN dataset is lower than for the X dataset (see Sect. 5.2). For both corpora, we did not observe a correlation between the length of a premise and its claim and the prediction quality of their relation.

One challenge in our analysis was the probabilistic nature of LLMs: Even for the same conversation and prompt, it may happen that the accuracy of the model changes considerably between runs. In order to achieve better stability between the runs, one could modify the prompts to include more specific constraints—potentially at the cost of generalization. This drawback could be mitigated by using specialized prompts for different types of posts.

6 Limitations

While our prompting strategies show promising results, there are still some limitations to consider. Due to rate limits and timeouts imposed by OpenAI, we had to apply chunking to the batched strategy, which may have affected the prediction quality. Additionally, we only consider the text of the posts and do not take into account other modalities like images or videos and thus are missing potentially valuable context information. We also do not analyze links that may be embedded in the posts. In the case of X posts, our current approach focuses on the replies to some starting post, but other relations like mentions or quoted tweets are not considered. Finally, an important aspect to consider is the runtime of the models. While predicting the polarities of a single conversation is a matter of seconds using the supervised model, the LLMs needed almost a minute to complete the task. The reason for this is that such generic LLMs have billions of parameters, while the smaller S-BERT model has millions only. We expect this to change in the future—even a model like S-BERT was considered to be too slow for use in production just a few years ago.

7 Conclusion and Future Work

In this paper, we presented an unsupervised approach to perform AM on posts from social networks. We introduced multiple prompting strategies for different context lengths and evaluated them on three different datasets. Our results show that the batched prompting strategy—when paired with an adequate LLM—is capable of matching or exceeding the effectiveness of a supervised LLM. Combined with our open-source implementation, this makes it possible to perform real-time AM on social networks even for emerging topics without appropriate training data.

In future work, the presented approach could be extended to also handle the classification of argumentative vs. non-argumentative posts. By adding a neutral class, posts that have little or no argumentative content could be detected and removed from the conversation tree. This could help boost the prediction accuracy, especially for datasets like the HN one where we currently need human annotators to do the job. Another interesting avenue for future work is the evaluation of the LLM Grok developed by xAI. Since this model is specifically trained on posts from X, we expect it to be more effective for this type of data than the generic LLMs used in this paper.

Acknowledgements. This work has been funded by the Deutsche Forschungsgemeinschaft (DFG) within the projects *ReCAP* and *ReCAP-II* (№ 375342983, 2018–2024) as part of the priority program *RATIO* (Robust Argumentation Machines, SPP-1999) as well as the *Studienstiftung*.

A Prompting Templates

A.1 Isolated Prompting

System You are a helpful assistant that predicts the relation/polarity between the premise and the claim of an argument. You shall predict whether the premise supports or attacks the claim. Answer either `support` or `attack`.
User Premise: `premise`. Claim: `claim`.

A.2 Sequential Prompting

System You are a helpful assistant that predicts the relation/polarity between the premise and the claim of an argument. You shall predict whether the premise supports or attacks the claim. Answer either `support` or `attack`.
User Premise: `premise`. Claim: `claim`.
Assistant `support` *or* `attack`
User Premise: `premise`. Claim: `claim`. And so on...

A.3 Contextualized Prompting

System You are a helpful assistant that predicts the relation/polarity between the premise and the claim of an argument. You shall predict whether the premise supports or attacks the claim. Answer either `support` or `attack`**User** Premise: `premise`. Claim: `claim`. The premise and the claim have the following neighbors in the conversation: `adu_1` ... `adu_n`

A.4 Batched Prompting

System You are a helpful assistant that predicts the relation/polarity between the premise and the claim of an argument. You shall predict whether the premise supports or attacks the claim. Answer either `support` or `attack`. You will be presented with a list of premise-claim pairs containing their text and id encoded as a JSON array. Provide exactly one prediction for each of them using the function `predict_entailment`.

Available Function Calls JSON schema describing `predict_entailment` as an array of objects with the following keys: `premise_id` (string), `claim_id` (string), and `polarity_type` (enum: support/attack).

User JSON array of objects with the following keys: `premise_id` (string), `premise_text` (string), `claim_id` (string), and `claim_text` (string).

References

1. Agarwal, V., Young, A.P., Joglekar, S., Sastry, N.: A graph-based context-aware model to understand online conversations. ACM Trans. Web **18**(1), 1–27 (2023)
2. Allen, J.F.: Natural language processing. In: Encyclopedia of Computer Science (2003)
3. Bosc, T., Cabrio, E., Villata, S.: DART: a dataset of arguments and their relations on Twitter. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016) (2016)
4. Bosc, T., Cabrio, E., Villata, S.: Tweeties squabbling: positive and negative results in applying argument mining on social media. In: Computational Models of Argument (2016)
5. Cabrio, E., Villata, S.: Detecting bipolar semantic relations among natural language arguments with textual entailment: a study. In: Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora (2013)
6. Cabrio, E., Villata, S.: A natural language bipolar argumentation approach to support users in online debate interactions. Argum. Comput. **4**(3), 209–230 (2013)
7. Cayrol, C., Lagasquie-Schiex, M.C.: On the acceptability of arguments in bipolar argumentation frameworks. In: Symbolic and Quantitative Approaches to Reasoning with Uncertainty (2005)
8. Chen, E., Deb, A., Ferrara, E.: #Election2020: the first public Twitter dataset on the 2020 US Presidential election. J. Comput. Soc. Sci. (2021)
9. Chesñevar, C.I., et al.: Towards an argument interchange format. Knowl. Eng. Rev. **21**(4), 293–316 (2006)

10. Cohen, J.: A coefficient of agreement for nominal scales. Educ. Psychol. Measur. **20**(1), 37–46 (1960)
11. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 (2018)
12. Dusmanu, M., Cabrio, E., Villata, S.: Argument mining on Twitter: arguments, facts and sources. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (2017)
13. Goudas, T., Louizos, C., Petasis, G., Karkaletsis, V.: Argument extraction from news, blogs, and social media. In: Artificial Intelligence: Methods and Applications (2014)
14. Gurevych, I., Lippi, M., Torroni, P.: Argumentation in social media. ACM Trans. Internet Technol. **17**(3), 1–2 (2017)
15. Karami, A., Lundy, M., Webb, F., Dwivedi, Y.K.: Twitter and research: a systematic literature review through text mining. IEEE Access **8**, 67698–67717 (2020)
16. Korman, D.Z., Mack, E., Jett, J., Renear, A.H.: Defining textual entailment. J. Assoc. Inf. Sci. Technol. **69**(6), 763–772 (2018)
17. Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. Biometrics (1977)
18. Lawrence, J., Reed, C.: Argument mining: a survey. Comput. Linguist. **45**(4), 765–818 (2019)
19. McNemar, Q.: Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika **12**(2), 153–157 (1947)
20. Ouertatani, A., Gasmi, G., Latiri, C.: Parsing argued opinion structure in Twitter content. J. Intell. Inf. Syst. **56**, 327–353 (2021)
21. Peldszus, A., Stede, M.: From argument diagrams to argumentation mining in texts - a survey. IJCINI **7**(1), 1–31 (2013)
22. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (2019)
23. Shevtsov, A., Oikonomidou, M., Antonakaki, D., Pratikakis, P., Ioannidis, S.: Analysis of Twitter and YouTube during USelections 2020. arXiv:2010.08183 (2020)
24. Stab, C., Gurevych, I.: Identifying argumentative discourse structures in persuasive essays. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014)
25. Stab, C., Gurevych, I.: Parsing argumentation structures in persuasive essays. Comput. Linguist. **43**(3), 619–659 (2017)
26. Touvron, H., et al.: Llama 2: Open Foundation and Fine-Tuned Chat Models (2023)
27. Wang, S., Fang, H., Khabsa, M., Mao, H., Ma, H.: Entailment as Few-Shot Learner. arXiv:2104.14690 (2021)

Argument Acquisition, Annotation and Quality Assessment

Are Large Language Models Reliable Argument Quality Annotators?

Nailia Mirzakhmedova(✉), Marcel Gohsen, Chia Hao Chang, and Benno Stein

Bauhaus-Universität Weimar, Weimar, Germany
{nailia.mirzakhmedova,marcel.gohsen,chiahao.chang,
benno.stein}@uni-weimar.de

Abstract. Evaluating the quality of arguments is a crucial aspect of any system leveraging argument mining. However, it is a challenge to obtain reliable and consistent annotations regarding argument quality, as this usually requires domain-specific expertise of the annotators. Even among experts, the assessment of argument quality is often inconsistent due to the inherent subjectivity of this task. In this paper, we study the potential of using state-of-the-art large language models (LLMs) as proxies for argument quality annotators. To assess the capability of LLMs in this regard, we analyze the agreement between model, human expert, and human novice annotators based on an established taxonomy of argument quality dimensions. Our findings highlight that LLMs can produce consistent annotations, with a moderately high agreement with human experts across most of the quality dimensions. Moreover, we show that using LLMs as additional annotators can significantly improve the agreement between annotators. These results suggest that LLMs can serve as a valuable tool for automated argument quality assessment, thus streamlining and accelerating the evaluation of large argument datasets.

Keywords: Argumentation quality · Automated argument quality assessment · Large language models · Argument mining

1 Introduction

Computational argumentation is an interdisciplinary research field that combines natural language processing with other disciplines such as artificial intelligence. A central question in computational argumentation is: What makes an argument good or bad? Depending on the goal of the author of a text, argument quality can involve a variety of dimensions. Evaluating the quality of an argument across these diverse dimensions demands a deep understanding of the topic at hand, often coupled with expertise from the argumentation literature. Hence, manual assessment of argument quality is a challenging and time-consuming process.

A promising technology to streamline argument quality assessment are large language models (LLMs) which have demonstrated impressive capabilities in

Supported by the German Research Foundation (Project Nr. 455911521).

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 129–146, 2024.
https://doi.org/10.1007/978-3-031-63536-6_8

tasks that require a profound understanding of semantic nuances and discourse structures. LLMs have been effectively employed in tasks such as summarization [32], question answering [16], and relation extraction [31]. Previous research has also investigated the usefulness of LLMs in argument mining tasks such as argument component identification [12], evidence detection [15], and stance classification [4]. Moreover, an emerging trend highlights the adoption of LLMs for data annotation purposes, such as sentiment analysis [8,24], relevance judgement [11], and harm measurement [18]. To the best of our knowledge, no prior work has investigated the potential of LLMs as annotators of argument quality.

In this paper, we analyze the reliability of LLMs as argument quality annotators by comparing automatic quality judgements with human annotations from both experts and novices.[1] We compare these quality ratings not only at an aggregate level, but also examine the individual components that make up argument quality. This includes looking at how well models can judge the relevance and coherence of an argument, the sufficiency of its evidential support, and the effectiveness of its rhetorical appeal. Ultimately, our objective is to understand whether LLMs can serve as a practical and reliable tool that supports and enhances human-led effort in argument quality assessment.

Specifically, we ask the following research questions regarding the potential of employing LLMs as argument quality annotators:

RQ1: Do LLMs provide more consistent evaluations of argument quality compared to human annotators?
RQ2: Do the assessments of argument quality made by LLMs align with those made by either human experts or human novices?
RQ3: Can integrating LLM annotations with human annotations significantly improve the resulting agreement in argument quality ratings?

In the following, Sect. 2 reviews the related work, Sect. 3 describes the experimental setup, including the dataset, the annotation procedure, and the employed models, and Sect. 4 presents the results of these experiments.

2 Related Work

We first review existing literature related to the evaluation and annotation of argument quality. Following that, we explore the works that examined the capabilities of large language models (LLMs) as data annotators as well as the degree of alignment between LLMs and human annotators.

2.1 Evaluating Argument Quality

Collecting argument quality annotations is an intricate task that often requires domain-specific knowledge, a number of annotators, and assured consistency in annotator reliability. Numerous works have studied argumentation quality across

[1] Code and data are available at github.com/webis-de/RATIO-24.

different domains, employing multiple annotators to classify and evaluate arguments based on various quality criteria. Park and Cardie [22] studied argumentation quality in the domain of web discourse. They employed two annotators to classify 9,000 web-based propositions into four categories based on their level of *verifiability.* Habernal and Gurevych [13] let five crowd-workers annotate a dataset consisting of 16,000 pairs of arguments with a binary *"is more convincing"* label, providing explanations for their decisions. Toledo et al. [27] collected a dataset of 14,000 pairs of arguments, each annotated with relative argument quality scores ranging from 0 to 1. They employed between 15 and 17 annotators for each instance to enhance the reliability of the collected annotations.

In the domain of student essays, Persing and Vincent [23] instructed six human annotators to evaluate 1,000 essays based on the *strength* of argumentation on a scale from 1 to 4. Carlile et al. [3] considered *persuasiveness* as the most important quality dimension of an argumentative essay. They asked two native English speakers to annotate 102 essays with argument components, argument persuasiveness scores, and further attributes such as *specificity, evidence, eloquence, relevance*, and *strength*, that determine the persuasiveness of an argument. Moreover, Marro et al. [19] employed three expert annotators for the annotation of essay components of Stab and Gurevych [25] for three basic argument quality dimensions: *cogency, rhetoric*, and *reasonableness.*

Aiming to create a unified understanding of argument quality properties, Wachsmuth et al. [30] proposed a comprehensive taxonomy of 15 argument quality dimensions derived from the argumentation literature. Three expert annotators were employed to annotate 320 arguments [13]. In Sect. 3, we use their quality annotations from 1 (low) to 3 (high) as a reference for our experiments.

Despite the multiple attempts and methodologies to evaluate argument quality, the process remains labor-intensive, time-consuming, and requires a significant degree of expertise. To facilitate the task of argument quality annotation, we propose employing LLMs, as they can potentially provide more reliable and consistent annotations while significantly reducing the required manual effort.

2.2 LLMs as Annotators

Recent work has expanded the role of LLMs from language generation and explored the potential of using LLMs as data annotators. Ding et al. [8] assessed the performance of GPT-3 [2] as a data annotator for sentiment analysis, relation extraction, named entity recognition, and aspect sentiment triplet extraction. They compared the efficiency of BERT [7], trained using data annotated by GPT-3, against BERT trained with human-annotated data. Their findings showed a noticeably similar performance level with substantially reduced annotation costs, promising a potentially cost-effective alternative in using GPT-3 for annotation. A study by Gilardi et al. [11] cross-examined the annotations by ChatGPT [20] and those by crowd-workers against expert annotations across four tasks: content relevance assessment, stance detection, topic detection, and general frame detection. They found that ChatGPT not only outperforms crowd-workers in terms of accuracy, but also shows a high degree of consistency in annotations.

The study by Gao et al. [10] explored automatic human-like evaluations of text summarization using ChatGPT compared to human experts. The model was prompted to evaluate the quality of summaries based on *relevance*, *coherence*, *fluency*, and *consistency* of the generated summaries. The authors found that ChatGPT's evaluations were highly correlated with those of human experts.

Zhuo et al. [35] proposed to use LLMs as evaluators of code generation. The authors used the CoNaLa dataset [34] and reported high example-level and corpus-level Kendall-Tau, Pearson, and Spearman correlations with human-rated code usefulness for various programming languages.

In the domain of information retrieval, Faggioli et al. [9] investigated the performance of GPT-3.5 and YouChat for query-passage relevance judgements. Given the high subjectivity of the task, their results showed a reasonable correlation between highly-trained human assessors and fully automated judgements.

Closest to our work is that by Chiang et al. [5], who compared the judgments of GPT-3 on text quality to expert human judgments on a 5-point Likert scale for four quality attributes: *grammaticality, cohesiveness, likability*, and *relevance*. Their findings revealed varying degrees of positive correlations between GPT-3 and human judgments, ranging from weak to strong.

When compared to existing research, our work pioneers the study of argument quality annotations generated by LLMs. In order to provide a thorough evaluation, we use an inter-annotator agreement metric to assess the consistency of annotations from these models, human experts and novices. This comparison allows us to understand the alignment between LLMs and human annotators, and to determine the potential of using LLMs as argument quality annotators.

3 Experimental Design

To investigate the reliability of large language models (LLMs) as annotators of argument quality, we conduct an experiment comparing human annotations with ratings generated automatically by LLMs. We treat LLMs as separate annotators and analyze the agreement both within and across groups of humans and models.

3.1 Expert Annotation

The goals of argumentation are manifold and include persuading audiences, resolving disputes, achieving agreement, completing inquiries, or recommending actions [26]. Due to the variety of these goals, the dimensions of argument quality are equally diverse. Based on a comprehensive survey of argumentation literature, Wachsmuth et al. [30] proposed a fine-granular taxonomy of argument quality dimensions that differentiates logical, rhetorical, and dialectic aspects. An overview of all quality dimensions is provided in Table 1.

In their work, Wachsmuth et al. [30] employed experts to rate the quality of arguments according to their proposed taxonomy. Three experts were selected out of a pool of seven based on their agreement in a pilot annotation study. These three experts comprised two PhDs and one PhD student (two female, one male) from

Table 1. Descriptions of argument quality dimensions as per Wachsmuth et al. [29].

Quality Dimension	Description
Cogency	Argument has (locally) acceptable, relevant, and sufficient premises
Local acceptability	Premises worthy of being believed
Local relevance	Premises support/attack conclusion
Local sufficiency	Premises enough to draw conclusion
Effectiveness	Argument persuades audience
Credibility	Makes author worthy of credence
Emotional appeal	Makes audience open to arguments
Clarity	Avoids deviation from the issue, and uses correct and unambiguous language
Appropriatness	Language proportional to the issue, supports credibility and emotions
Arrangement	Argues in the right order
Reasonableness	Argument is (globally) acceptable, relevant, and sufficient
Global acceptability	Audience accepts use of argument
Global relevance	Argument helps arrive at agreement
Global sufficiency	Enough rebuttal of counterarguments
Overall quality	Argumentation quality in total

three different countries. To construct the Dagstuhl-15512-ArgQuality corpus, the selected experts annotated 320 arguments from the UKPConvArgRank dataset [13]. The resulting corpus contains 15 quality dimensions for each argument, each rated on a 3-point Likert scale (low, medium, high) or as not assessable. Each argument in the corpus belongs to one of 16 different topics and takes a stance for or against the topic. The dataset is balanced and contains 10 supporting and 10 attacking arguments per topic. The annotation guidelines, which define all quality dimensions in more detail, are publicly available online.[2]

3.2 Novice Annotation

To provide an additional point of reference for determining the abilities of LLMs as argument quality annotators, we conducted an annotation study involving humans with no prior experience with computational argumentation. We asked undergraduate students to assess the quality of arguments from the Dagstuhl-15512-ArgQuality corpus using the same taxonomy.

The expert annotation guidelines require that annotators have expertise in computational argumentation. To make this task accessible for novices, we paraphrased the annotation guidelines and the definitions of argument quality dimen-

[2] https://zenodo.org/records/3973285

sions to ensure clarity and comprehension. These simplified definitions for each quality dimension can be found in the Appendix. To illustrate, the expert definition of *local acceptability* of an argument is stated as follows:

Definition 1 (Local Acceptability (Expert)). *A premise of an argument should be seen as acceptable if it is worthy of being believed, i.e., if you rationally think it is true or if you see no reason for not believing that it may be true.*

The above definition requires an annotator to distinguish between premises and arguments. To ease the understanding and reduce the necessary prior knowledge, we simplify the definition of local acceptability as follows:

Definition 2 (Local Acceptability (Novice)). *The reasons are individually believable: they could be true.*

We refer to arguments as "reasons" within the simplified guidelines and combine the stance with the issue into a "conclusion". For example, given the issue *"Is TV better than books?"* and the stance *"No it isn't"*, we state the conclusion as *"TV is not better than books"*.

Each novice annotator was presented with an argument, a conclusion, and the simplified definitions of the quality dimensions. Identical to the annotation procedure for expert annotations, the annotators were tasked to rate each quality dimension of the argument on a 3-point Likert scale or as not assessable.

In total, we acquired 108 students to annotate the quality of the 320 arguments from the dataset. We assigned 10 arguments to each student to annotate in order to obtain at least three annotations per argument and quality dimension. Since not all students finished their annotations and some students annotated a wrong set of arguments, we obtained a minimum of three annotations per argument and quality dimension only for 248 arguments. We treat the missing annotations of the 72 arguments as non-evaluable. For the 163 arguments for which we collected more than 3 annotations, we select three annotations that maximize the inter-annotator agreement measured by Krippendorff's α.

3.3 Models

Due to the complexity of the task, we focus on state-of-the-art LLMs for the automatic evaluation of argumentation quality. Building upon previous research regarding LLMs as annotators (cf. Sect. 2.2), one of the most commonly used models is GPT-3 [2]. Specifically, we use the `gpt-3.5-turbo-0613` accessible via OpenAI's API.[3] Despite the availability of the newer GPT-4 model [21], we do not employ it in our study due to the significantly higher associated costs.

In addition, we use Google's recently released PaLM 2 model [1], the successor to the original PaLM model [6]. The authors report comparable results to GPT-4 in semantic reasoning tasks, which makes it interesting for the evaluation of argument quality. For PaLM 2, we use the `text-bison@001` version of the model.

[3] https://platform.openai.com/

```
### Instruction:
Please answer the following questions for the given comment from
an online debate forum on a given issue.

### Issue:
Is TV better than books?

### Stance:
No, it isn't.

### Argument:
Books will be always great whatever the new technological
developments emerges books has its fixed place in every humans
heart.

### Quality dimension definition:
Clarity: The style of an argumentation should be seen as clear if
it uses grammatically correct and widely unambiguous language as
well as if it avoids unnecessary complexity and deviation from the
discussed issue. The used language should make it easy for you to
understand without doubts what the author argues for and how.

### Question:
How would you rate the clarity of the style of the author's
argumentation? Choose one of the options below [and explain your
reasoning]:
3 - High
2 - Medium
1 - Low
? - Cannot judge
```

Fig. 1. An expert prompt that contains instructions and an example issue, stance, and argument from the Dagstuhl-15512 ArgQuality corpus. This particular prompt example asks the model to rate the *clarity* of the argument. The reasoning variant of this prompt is colored in gray.

Both PaLM 2 and GPT-3 are closed-source language models. We initially intended to incorporate Meta's Llama 2 model [28] in our experiments, in order to evaluate the performance of open-source LLMs on our task. However, in pilot experiments, Llama 2 with 7 billion parameters did not follow the instructions and therefore did not produce quality scores. Even though the 13 billion parameter version of Llama 2 did generate quality scores, they were seemingly random, with agreement across the multiple runs close to zero. Due to hardware limitations, we did not test the largest Llama 2 model with 70 billion parameters.

PaLM 2 and GPT-3 allow to specify a set of parameters such as temperature to control the diversity of the output, where lowering the temperature reduces the 'randomness' of the output. For our experiments, we choose a reasonably low temperature of 0.3. Other parameters that we keep constant across models include

$p = 1.0$ of the nucleus sampling [14], most probable tokens $k = 40$, and a maximum of 256 newly generated tokens.

3.4 Prompting

Two different groups of human annotators, the expert annotators of Wachsmuth et al. [30] and the novice annotators recruited for this work, had access to different knowledge sources in their annotation guidelines. To determine whether the impact of this difference is similar between humans and LLMs, we created prompts that reflect the knowledge from the annotation guidelines of experts and novices. We refer to these prompt types as *expert* and *novice* prompts.

Besides instructions, an expert prompt consists of an issue, a stance, and an argument from the Dagstuhl-15512-ArgQuality corpus. The expert prompt also contains the name and original definition of the quality dimension from Wachsmuth et al. [30] as well as the annotation scheme (3-point Likert scale or "not assessable"). An example of an expert prompt is shown in Fig. 1.

In contrast to the expert prompt type, novice prompts contain a conclusion (as described in Sect. 3.2) instead of an issue and stance. In the novice prompt, the definition of the quality dimension to be assessed is replaced by the simplified definition. However, the textual argument, which is renamed to "reasons", and the annotation scheme remain identical to the expert prompt.

Recently, it has been shown that explanation-augmented prompts can elicit reasoning capabilities in LLMs and improve their performance across various tasks [17,33]. In pilot experiments, we found that GPT-3 produces more consistent annotations if we prompt the model to provide an explanation for the chosen score. We therefore test *reasoning* prompt variants in which we ask the model to provide an explanation for the generated quality rating.

To take the randomness of the output of LLMs for the same prompt into account, each prompt variant is repeated (at least) three times for each argument and quality dimension. Each prompt repetition is considered as a separate annotator in order to calculate the agreement between the annotations and to draw conclusions about the consistency of the quality annotations of LLMs.

4 Results

To understand the strengths and weaknesses of large language models (LLMs) as argument quality assessors and to answer our research questions, we use the prompting approaches described in Sect. 3.4 to generate LLM annotations for arguments from the Dagstuhl-15512-ArgQuality corpus. The dataset contains 320 statements, 16 of which were originally judged as non-argumentative by expert human annotators and therefore are excluded from the analysis.

First, to identify biases in human and LLM argument quality annotations, we analyze the distribution of assigned labels across all quality dimensions. This distribution is visualized in Fig. 2. Human annotations show an almost balanced distribution between low, medium and high quality ratings. However, it is noteworthy that human novices show a tendency to assign high ratings more frequently

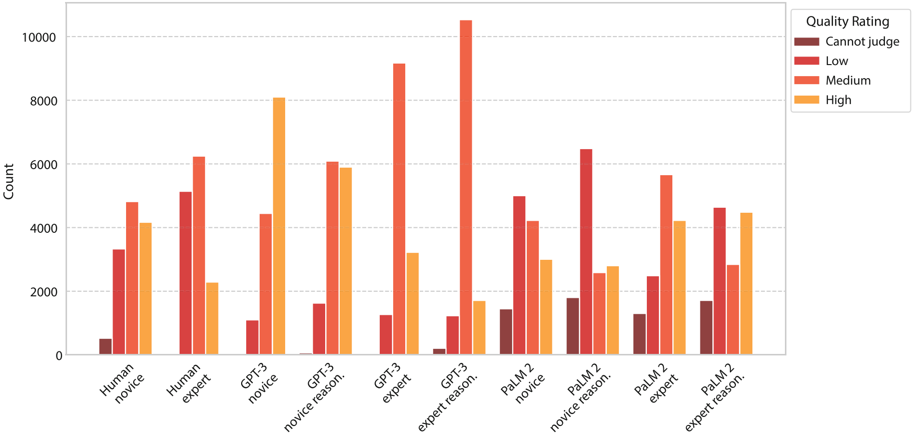


Fig. 2. Distribution of the assigned quality ratings across all quality dimensions compared between human annotators and LLMs.

than experts. As for models, GPT-3 with expert prompts displays a much more skewed distribution, showing a strong bias towards medium ratings, deviating from the trend observed in human assessors. On the contrary, when GPT-3 is prompted with novice-level guidelines, it tends to assign high-quality ratings more frequently. Notably, annotations generated by PaLM 2 have a similar distribution to that of human annotators which seems promising for the subsequent analysis of agreement with human assessments.

Overall, it can be stated that not only the choice of model, but also the prompt type has a major influence on the generated argument quality ratings. Even slight prompt modifications, such as asking to justify the score, can result in a notable change in the assigned quality scores, which is especially prominent for PaLM 2 with expert prompts in our case. Another interesting observation is that GPT-3 almost always provides a rating for a given dimension: only in 214 out of 21,120 cases ($\approx$1%) this model did not generate a score. The instances where PaLM 2 did not assess argument quality sum up to 4,972 ($\approx$23%) and mostly stem from content policies, particularly in cases where arguments revolve around graphic topics such as pornography or contain offensive statements.

4.1 Consistency of Argument Quality Annotations

We address our first research question concerning the consistency of argument quality assessments by comparing the agreement levels within LLM groups with those of human assessors. To quantify the agreement within each group of annotators, we use Krippendorff's α. To ensure a fair comparison with human annotators, we evaluate the agreement between three LLM annotation runs.

Table 2 shows Krippendorff's α for human experts, human novices, and all LLM prompt variants across individual quality dimensions as well as overall agreement. Human annotators exhibit generally low agreement, with a maximum of

Table 2. Inter-annotator agreement per argument quality dimension within each group of human annotators and LLMs, reported as Krippendorf's α. The dimension with the highest agreement within each group is marked in bold.

Quality Dimension	Human		GPT-3				PaLM 2			
	Novice	Expert	Novice	Expert	Reasoning		Novice	Expert	Reasoning	
					Novice	Expert			Novice	Expert
Cogency	0.38	0.38	0.72	0.73	0.77	0.72	**0.99**	0.98	0.73	0.74
Local Acceptability	**0.43**	0.33	0.64	0.69	0.70	0.75	0.98	0.97	0.60	0.71
Local Relevance	0.36	0.41	0.70	0.59	0.76	0.61	0.98	0.98	0.78	0.68
Local Sufficiency	0.35	0.27	0.74	0.69	0.79	0.72	0.98	0.97	0.63	0.63
Effectiveness	0.41	0.33	0.72	0.70	0.77	0.74	0.98	**0.99**	0.78	0.79
Credibility	0.36	0.23	**0.79**	**0.79**	0.81	0.78	**0.99**	0.97	0.72	0.67
Emotional Appeal	0.35	0.21	0.73	0.56	0.74	0.70	0.98	0.97	0.64	0.72
Clarity	0.27	0.25	0.72	0.69	0.71	0.69	**0.99**	**0.99**	**0.82**	0.80
Appropriateness	0.39	0.17	0.66	0.50	0.68	0.58	**0.99**	**0.99**	0.75	0.81
Arrangement	0.39	0.26	0.68	0.66	0.71	0.69	**0.99**	**0.99**	0.69	0.65
Reasonableness	0.35	**0.45**	0.73	0.78	0.81	0.74	0.97	0.97	0.70	0.76
Global Acceptability	0.37	0.39	0.72	0.77	0.77	0.74	0.98	0.97	0.66	0.70
Global Relevance	0.38	0.26	0.69	0.71	0.81	0.70	**0.99**	0.98	0.74	**0.85**
Global Sufficiency	0.27	0.17	0.72	0.69	0.72	0.75	0.98	0.96	0.62	0.47
Overall Quality	0.41	0.44	0.77	0.77	**0.82**	**0.81**	0.98	0.97	0.77	0.78
Overall α	0.37	0.40	0.76	0.73	0.78	0.74	0.99	0.98	0.76	0.78

0.43 on the *local acceptability* dimension for novices and 0.45 on *reasonableness* for experts. This low level of agreement between humans emphasizes the subjectivity and complexity of assessing argument quality in a fine-grained taxonomy. For most of the quality dimensions, novice annotators show slightly higher agreement than those of experts, which could be due to the clearer definitions of the quality dimensions or perhaps due to the optimization of agreement for arguments that received more than three annotations (cf. Sect. 3.2).

In contrast, LLM agreement between annotation repetitions is substantially higher. Interestingly, the PaLM 2 model shows near-perfect agreement for both expert and novice prompts, but shows a notable drop when asked to explain its reasoning. In contrast to PaLM 2, the GPT-3 model exhibits a slight improvement in agreement when asked to provide an explanation. Such disparities might be due to the differences in the underlying architectures and training methodologies of the two models, which require further exploration beyond the work at hand. Overall, both models show a high degree of agreement across different runs, with varying impact of reasoning prompts on the agreement depending on the employed model.

RQ1: Do LLMs provide more consistent evaluations of argument quality compared to human annotators? The observed low agreement among human annotators underscores that evaluating argument quality is indeed a subjective and challenging task. In contrast, the significantly higher agreement among different LLM runs highlights the potential of these models for providing more consistent argument quality annotations.

Table 3. Number of arguments with perfect agreement for each argument dimension within each group of human annotators (expert, novice).

Quality Dimension	Expert	Novice
Cogency	122	105
Local Acceptability	82	115
Local Relevance	99	100
Local Sufficiency	113	89
Effectiveness	128	118
Credibility	115	86
Emotional Appeal	130	90
Clarity	89	92
Appropriateness	53	102
Arrangement	81	102
Reasonableness	126	119
Global Acceptability	96	102
Global Relevance	66	96
Global Sufficiency	136	81
Overall Quality	134	130

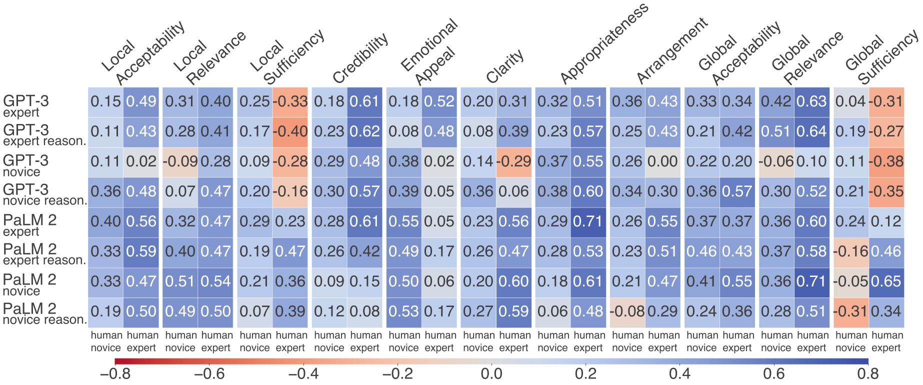


Fig. 3. Inter-annotator agreement (Krippendorff's α) between human and LLM annotations for each fine-grained argument quality dimension.

4.2 Agreement Between Humans and LLMs

We discovered that LLMs generate annotations more consistently than humans. However, to assert that LLMs can reliably evaluate the quality of arguments, we need to test how the automatic annotations align with the human annotations. Given the low agreement among human annotators, we created subsets of arguments for each quality dimension, where either all expert annotators or all

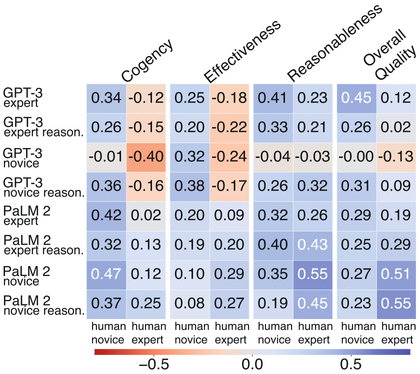


Fig. 4. Inter-annotator agreement (Krippendorff's α) between human and LLM annotations for each coarse-grained argument quality dimension.

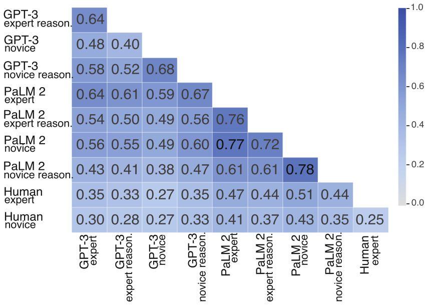


Fig. 5. Overall inter-annotator agreement (Krippendorff's α) between each combination of human expert, novice, and LLM-generated annotations.

novice annotators unanimously agreed on a score. Table 3 presents the statistics of the resulting subsets with perfect agreement, which we employ for further inter-annotator agreement analysis.

Figure 3 shows the agreement for each quality dimension, as measured by Krippendorff's α, between human annotations and automatically generated quality ratings by LLMs with different prompts. Overall, we observe moderate agreement across most quality dimensions, with the annotations by PaLM 2 reaching a maximum of 0.71 for *appropriateness* and *global relevance*. Regardless of the prompt type, PaLM 2 annotations generally achieve higher agreement with human annotations compared to GPT-3. In the case of *local* and *global sufficiency*, there are even systematic disagreements between the GPT-3 assessments and those of human experts. Similarly, disagreement is observed between PaLM 2 annotations and human novices for the *global sufficiency* dimension.

Overall, there is a large variance in agreement between model and human judgments across different quality dimensions. For example, while the agreement on *credibility* and *appropriateness* is in the range of [0.08, 0.62] and [0.06, 0.71] respectively, the agreement on *local* and *global sufficiency* fluctuates even more.

In terms of prompt variants, we can see that GPT-3 with expert prompts shows a higher agreement with human expert annotations than with human novice annotations, and a similar trend is observed for GPT-3 with novice prompts and human novices. On the other hand, PaLM 2 with either of the prompt types tends to show higher agreement with human experts. Similar findings can be inferred from the agreement between LLMs and human novices and experts on the coarse-grained quality dimensions that are visualized in Fig. 4.

RQ2: Do the assessments of argument quality made by LLMs align with those made by either human experts or human novices? We found that LLMs agree most with human argument quality ratings on fine-grained quality dimensions such as *credi-*

Table 4. Change in overall Krippendorf's α after adding LLM annotations to human expert or novice annotations. Significant changes ($p < 0.05$) between the agreement of the original annotations and the modified annotations set are marked with *.

Annotations	Expert		Novice		Annotations	Expert		Novice	
	GPT-3	PaLM 2	GPT-3	PaLM 2		GPT-3	PaLM 2	GPT-3	PaLM 2
Human experts	0.40	0.40	0.40	0.40	Human novices	0.37	0.37	0.37	0.37
+1 annotation	0.32*	0.37*	0.30*	0.37*	+1 annotation	0.27*	0.29*	0.27*	0.27*
+2 annotations	0.31*	0.40	0.32*	0.40	+2 annotations	0.26*	0.33	0.29*	0.30
+3 annotations	0.33	0.44*	0.35	0.44*	+3 annotations	0.28*	0.37*	0.33	0.35*
+4 annotations	0.35	0.47*	0.39	0.47*	+4 annotations	0.30*	0.41*	0.37	0.39*
+5 annotations	0.37	0.50*	0.42*	0.50*	+5 annotations	0.32*	0.45*	0.40*	0.43*

bility, *emotional appeal*, *appropriateness*, and *global relevance* or on coarse-grained dimensions such as *reasonableness* and *overall quality*. Overall, we found varying degrees of agreement between LLMs and human annotators, with PaLM 2 annotations tending to generally align more with those of humans.

4.3 LLMs as Additional Annotators

LLMs can be employed either as independent automatic argument quality raters or as a source of additional annotations to validate a set of human annotations. For the second scenario, we analyze the overall agreement between different combinations of human (expert or novice) and LLM annotators.

Figure 5 illustrates the overall Krippendorff's α agreement for each combination of annotator groups. We can see that there is low to medium agreement for each combination of annotators, with the lowest value being 0.25 between human novices and human experts and the highest value being 0.77 between PaLM with novice and expert prompts. Regardless of the prompt type, the agreement between PaLM 2 and GPT-3 is moderate, ranging from 0.38 to 0.67. This suggests the potential efficacy of employing diverse models as supplementary annotators.

We further investigate whether the agreement changes if we incrementally integrate automatically generated annotations into the original set of human annotations. The results reported in Table 4 show that adding PaLM 2 annotations can significantly improve the agreement of human experts as well as human novices. A significant increase is already visible after adding three annotations to the annotations of human experts and four to the annotations of human novices. However, the introduction of GPT-3 annotations leads to a significant decrease in agreement. This can be attributed to the relatively low level of agreement between GPT-3 and human annotators (cf. Fig. 5).

RQ3: Can integrating LLM annotations with human annotations significantly improve the resulting agreement in argument quality ratings? The analysis indicates that the impact on agreement levels when incorporating generated quality assessments with human annotations varies based on the employed LLM. When using a powerful model such as PaLM 2, the agreement of human annotations can be significantly increased by adding three or more generated annotations. These results underscore LLMs as valuable contributors to the annotator ensemble.

5 Conclusion

In this paper, we investigated the effectiveness of LLMs, specifically GPT-3 and PaLM 2, in evaluating argument quality. We utilized four distinct prompt types to solicit quality ratings from these models and compared their assessments with those made by human novices and experts. The results reveal that LLMs exhibit greater consistency in evaluating argument quality compared to both novice and expert human annotators, showcasing their potential reliability. Based on our empirical analysis, we can recommend two modes of application for LLMs as annotators of argument quality: (1) a fully automatic annotation procedure with LLMs as automatic quality raters, for which we found moderately high agreement between PaLM 2 and human expert quality ratings, or (2) a semi-automatic procedure using LLMs as additional quality annotators, resulting in a significant enhancement in agreement when combined with human annotations. In both Modi, LLMs can serve as a valuable tool for streamlining the argument quality annotation process on a large scale.

To further minimize annotation expenses, we intend to expand these experiments to various open-source large language models. In addition to the investigated the zero-shot prompting technique, enhancing agreement with human annotations could involve utilizing few-shot prompting technique or fine-tuning LLMs based on human judgments of argument quality. We see great potential in LLMs as argument quality raters, which, if further optimized to agree more closely with human assessments, can reduce manual effort and expenses, establishing them as valuable tools in argument mining.

6 Limitations

The experiments in this paper are based on the hypothesis that multiple runs of the same model, prompt, and hyperparameters simulate different annotators as a result of nucleus sampling. This hypothesis has not yet been proven, and its validity cannot be inferred from the analysis. The higher inter-model agreement indicates a lower variance in the automatically generated annotations, which might argue against this hypothesis. Therefore, the agreement between model and human annotations has been calculated using examples with perfect agreement only in order to exclude effects of this variance. Further experiments are needed to determine how to replicate the behavior of different annotators. can be found in Appendix Although we deeply investigate LLMs as quality assessors for arguments, the generalizability of our results beyond argumentation is not yet clear. However, due to the complexity and subjectivity of argument quality assessment, as seen from the low human inter-annotator agreement, we argue that this task might be a worst-case scenario for LLMs, and we would expect comparable or even better results in less subjective task domains. However, more experiments are needed to confirm or reject this hypothesis.

Acknowledgements. This research has been supported by the German Research Foundation (DFG) with grant number 455911521, project “LARGA” in SPP “RATIO”.

Appendix

Table 5 lists the adapted definitions of argument quality dimensions employed for novice annotations.

Table 5. The set of simplified definitions of argument quality dimensions.

Quality Dimension	Definition
Local acceptability	The reasons are individually believable: they could be true
Local relevance	The reasons (assuming they are true) are relevant to the conclusion: they tell why one could accept the conclusion
Local sufficiency	The reasons (assuming they are true) are sufficient to draw the conclusion: no other reason is necessary to arrive at the conclusion
Credibility	The reasons make the author seem trustworthy and knowledgeable
Emotional appeal	The reasons can make people feel emotions that make them willing to agree with the author
Clarity	The author uses clear, grammatically correct and unambiguous language. The author sticks to the main topic and does not make things overly complicated
Appropriatness	The author uses an appropriate style for the reasons and this style fits to the topic’s importance
Arrangement	The reasons are properly organized: coherent, easy to follow, well-structured
Global acceptability	The reasons and conclusion combined are believable: everything taken together could be true
Global relevance	The reasons (assuming they are true) are relevant for resolving a discussion around the conclusion’s topic
Global sufficiency	The reasons (assuming they are true) are sufficient in that they consider any counter-arguments that may arise

References

1. Anil, R., Dai, A.M., Firat, O., Johnson, M., et al.: PaLM 2 Technical Report. CoRR abs/2305.10403 (2023)
2. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., et al.: Language models are few-shot learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, 6–12 December 2020, virtual (2020)
3. Carlile, W., Gurrapadi, N., Ke, Z., Ng, V.: Give me more feedback: annotating argument persuasiveness and related attributes in student essays. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, 15–20 July 2018, Volume 1: Long Papers, pp. 621–631. Association for Computational Linguistics (2018)
4. Chen, G., Cheng, L., Tuan, L.A., Bing, L.: Exploring the Potential of Large Language Models in Computational Argumentation. CoRR abs/2311.09022 (2023)
5. Chiang, D.C., Lee, H.: Can large language models be an alternative to human evaluations? In: Rogers, A., Boyd-Graber, J.L., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, 9–14 July 2023, pp. 15607–15631. Association for Computational Linguistics (2023)
6. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., et al.: PaLM: Scaling Language Modeling with Pathways. CoRR abs/2204.02311 (2022)
7. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, 2–7 June 2019, Volume 1 (Long and Short Papers), pp. 4171–4186. Association for Computational Linguistics (2019)
8. Ding, B., et al.: Is GPT-3 a good data annotator? In: Rogers, A., Boyd-Graber, J.L., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, 9–14 July 2023, pp. 11173–11195. Association for Computational Linguistics (2023)
9. Faggioli, G., et al.: Perspectives on large language models for relevance judgment. In: Yoshioka, M., Kiseleva, J., Aliannejadi, M. (eds.) Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2023, Taipei, Taiwan, 23 July 2023, pp. 39–50. ACM (2023)
10. Gao, M., Ruan, J., Sun, R., Yin, X., Yang, S., Wan, X.: Human-like Summarization Evaluation with ChatGPT. CoRR abs/2304.02554 (2023)
11. Gilardi, F., Alizadeh, M., Kubli, M.: ChatGPT Outperforms Crowd-Workers for Text-Annotation Tasks. CoRR abs/2303.15056 (2023)
12. Guo, J., Cheng, L., Zhang, W., Kok, S., Li, X., Bing, L.: AQE: argument quadruplet extraction via a quad-tagging augmented generative approach. In: Rogers, A., Boyd-Graber, J.L., Okazaki, N. (eds.) Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, 9–14 July 2023, pp. 932–946. Association for Computational Linguistics (2023)
13. Habernal, I., Gurevych, I.: Which argument is more convincing? Analyzing and predicting convincingness of web arguments using bidirectional LSTM. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, 7–12 August 2016, Berlin, Germany, Volume 1: Long Papers. The Association for Computer Linguistics (2016)

14. Holtzman, A., Buys, J., Forbes, M., Choi, Y.: The Curious Case of Neural Text Degeneration. CoRR abs/1904.09751 (2019)
15. Huo, S., Arabzadeh, N., Clarke, C.L.A.: Retrieving supporting evidence for generative question answering. In: Ai, Q., Liu, Y., Moffat, A., Huang, X., Sakai, T., Zobel, J. (eds.) Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region, SIGIR-AP 2023, Beijing, China, 26–28 November 2023, pp. 11–20. ACM (2023)
16. Kamalloo, E., Dziri, N., Clarke, C.L.A., Rafiei, D.: Evaluating open-domain question answering in the era of large language models. In: Rogers, A., Boyd-Graber, J.L., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, 9–14 July 2023, pp. 5591–5606. Association for Computational Linguistics (2023)
17. Kojima, T., Gu, S.S., Reid, M., Matsuo, Y., Iwasawa, Y.: Large language models are zero-shot reasoners. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, 28 November–9 December 2022 (2022)
18. Magooda, A., Helyar, A., Jackson, K., Sullivan, D., et al.: A Framework for Automated Measurement of Responsible AI Harms in Generative AI Applications. CoRR abs/2310.17750 (2023)
19. Marro, S., Cabrio, E., Villata, S.: Argumentation quality assessment: an argument mining approach. In: ECA 2022-European Conference on Argumentation (2022)
20. OpenAI: ChatGPT (2022)
21. OpenAI: GPT-4 Technical Report. CoRR abs/2303.08774 (2023)
22. Park, J., Cardie, C.: Identifying appropriate support for propositions in online user comments. In: Proceedings of the First Workshop on Argument Mining, hosted by the 52nd Annual Meeting of the Association for Computational Linguistics, ArgMining@ACL 2014, 26 June 2014, Baltimore, Maryland, USA, pp. 29–38. The Association for Computer Linguistics (2014)
23. Persing, I., Ng, V.: Modeling argument strength in student essays. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, 26–31 July 2015, Beijing, China, Volume 1: Long Papers, pp. 543–552. The Association for Computer Linguistics (2015)
24. Rønningstad, E., Velldal, E., Øvrelid, L.: A GPT among annotators: LLM-based entity-level sentiment annotation. In: Proceedings of the 18th Linguistic Annotation Workshop (LAW-XVIII), pp. 133–139 (2024)
25. Stab, C., Gurevych, I.: Parsing argumentation structures in persuasive essays. Comput. Linguist. **43**(3), 619–659 (2017)
26. Tindale, C.W.: Fallacies and Argument Appraisal, 1st edn. Cambridge University Press, Cambridge (2007)
27. Toledo, A., Gretz, S., Cohen-Karlik, E., Friedman, R., et al.: Automatic argument quality assessment - new datasets and methods. In: Inui, K., Jiang, J., Ng, V., Wan, X. (eds.) Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, 3–7 November 2019, pp. 5624–5634. Association for Computational Linguistics (2019)
28. Touvron, H., Martin, L., Stone, K., Albert, P., et al.: Llama 2: Open Foundation and Fine-Tuned Chat Models. CoRR abs/2307.09288 (2023)

29. Wachsmuth, H., et al.: Argumentation quality assessment: theory vs. practice. In: Barzilay, R., Kan, M.Y. (eds.) 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017), pp. 250–255. Association for Computational Linguistics (2017)
30. Wachsmuth, H., et al.: Computational argumentation quality assessment in natural language. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pp. 176–187 (2017)
31. Wadhwa, S., Amir, S., Wallace, B.C.: Revisiting relation extraction in the era of large language models. In: Rogers, A., Boyd-Graber, J.L., Okazaki, N. (eds.) Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, 9–14 July 2023, pp. 15566–15589. Association for Computational Linguistics (2023)
32. Wang, Y., Zhang, Z., Wang, R.: Element-aware Summarization with Large Language Models: Expert-aligned Evaluation and Chain-of-Thought Method. arXiv preprint arXiv:2305.13412 (2023)
33. Wei, J., et al.: Chain-of-thought prompting elicits reasoning in large language models. In: Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., Oh, A. (eds.) Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, 28 November–9 December 2022 (2022)
34. Yin, P., Deng, B., Chen, E., Vasilescu, B., Neubig, G.: Learning to mine aligned code and natural language pairs from stack overflow. In: Zaidman, A., Kamei, Y., Hill, E. (eds.) Proceedings of the 15th International Conference on Mining Software Repositories, MSR 2018, Gothenburg, Sweden, 28–29 May 2018, pp. 476–486. ACM (2018)
35. Zhuo, T.Y.: Large Language Models Are State-of-the-Art Evaluators of Code Generation. CoRR abs/2304.14317 (2023)

The Impact of Argument Arrangement on Essay Scoring

René Knaebel(✉), Robin Schaefer, and Manfred Stede

Applied Computational Linguistics, University of Potsdam, 14476 Potsdam, Germany
{rene.knaebel,robin.schaefer,stede}@uni-potsdam.de

Abstract. We study the question to what extent the task of predicting the quality of student essays can be supported with computing "flows" of semantic types of argumentative units. Specifically, we use tagsets for claim and premise types that were recently applied to the Argument Annotated Essays corpus (AAE; Stab/Gurevych 2017) by Schaefer et al (2023). We train argument component and semantic type classification models on AAE and then use them to label the essays in two corpora that have numeric essay ratings, viz. FEEDBACK/PERSUADE and ICLE. We train linear classification models on flow features and find that flows of our semantic types are a better predictor for essay quality (in a simplified, good/bad dichotomy) than flows of coarse argument components (major claim, claim, premise). Finally, we calculate feature impact and perform a qualitative inspection, which shows some tendencies for pattern occurrence in the two essay classes.

Keywords: Argument Mining · Essay Scoring · Argumentation Strategy

1 Introduction

Over the last decade, the field of Argument Mining (AM) has grown into a fruitful area of study that comprises a set of challenging sub-tasks [16,32]. In our work, we make use of the automatic identification and extraction of argument components, i.e., claims [8,25] and premises [23]. This has been studied for different text domains including news editorials [2], Wikipedia articles [23], social media data like tweets [27], and student essays [31]; the latter are the domain that we address here.

One application of analyzing argumentation in student essays is in contributing to assessing the quality of an essay. To this end, a variety of argument-related features have been studied and found to be useful in the past (see Sect. 2). In this paper, we add features of "flows" (sequences of occurrence in the text) of the types of claims and premises. We compare the impact of coarse types (major claim, claim, premise) to fine-grained semantic types of those components (e.g., *fact, value* and *policy* claims; see Sect. 3.1). We achieve this by utilizing the Argument Annotated Essays (AAE) corpus [31] for training ADU identification

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 147–162, 2024.
https://doi.org/10.1007/978-3-031-63536-6_9

and semantic type classification models. These models are used to automatically label our two target essay corpora Feedback [7] and ICLE [11], which have previously been annotated with essay quality ratings, with ADUs and their types. We then extract semantic type flows and use them as features in linear classification models for essay quality prediction.

Our two contributions are (i) the finding that for some dimensions of essay quality, flows of fine-grained features are more powerful predictors than flows of the coarse features; and (ii) a qualitative analysis that leads to some observations on correlations between flow patterns and essay quality.

The next section provides an overview of related work, and Sect. 3 introduces the three corpora we are working with, and the features we use for semantic types. In Sect. 4, we describe our experiments, which involve some "within-domain transfer" in that we train on an essay corpus annotated for the component features but that does not have quality scores [31] and then run those models on two corpora that offer scores but no (compatible) type annotation [7,11]. We discuss the findings in Sect. 5 and conclude in Sect. 6.

2 Related Work

Argument Mining in Essays. The *AAE* corpus, consisting of 402 essays with claims, premises and relations among them [31], is a widely-used resource for developing AM techniques. We mention a few, viz. component detection [30], semantic type annotation and identification [4,26], essay quality assessment [4,33], and end-to-end AM [21,24]. It was also applied in research on unsupervised AM [22], the analysis of argumentation strategies [26], and multi-scale AM [34]. The latter utilizes the text units *essay*, *paragraph*, and *word* for major claim, claim and premise identification, respectively. Another essay corpus that received attention in AM is ICLE [12]. For example, [5] used its rich annotations to compare aspects of argumentation strategies across different cultural groups among English learners.

Argument Component Types. Specific types of argument components have been used to label claims and premises in a variety of text genres. In Wikipedia [23], editorials [2], and persuasive essays [4,26] premises have been annotated as, e.g., *study/statistics*, *expert/testimony*, *anecdote* or *common knowledge/common ground*. Other annotated premise types include *study*, *factual*, *opinion*, and *reasoning* in idebate.org data [15]. For claims, *fact*, *value* and *policy* have been annotated in persuasive essays [4,26], in addition to *logos*, *pathos*, and *ethos* [4], i.e. Aristotle's modes of persuasion [14]. Claims in Amazon reviews have been labeled with the types *fact*, *testimony*, *policy*, and *value* [6].

Social media text has been a popular target, too. Annotated types include evidence types typical for social media, e.g. *news media accounts*, *blog posts*, or *pictures* [1], *factual* vs *opinionated* [9], and more recently *un/verifiability*, *reason* and *external/internal evidence* [27]. Furthermore, discussions collected from the subreddit Change My View were annotated for the claim types *interpretation*, *evaluation-rational*, *evaluation-emotional*, and *agreement/disagreement*, while premises were labeled with *logos*, *pathos*, and *ethos* [13].

In our work, we apply the set of claim and premise types that we described in our recent work on argument strategy analysis [26]. It was derived and extended from previous studies [2,4].

Argument Analysis for Essay Scoring. In early work, [18] found correlations between distributions of argument component types and holistic essay scores. In contrast, [29] evaluated the *contents* of the arguments in relation to the argument scheme present in the essay prompt. Building on their data, [3] turned to structure and found a moderate positive correlation between holistic essay scores and distributions of argument components and relations. Similarly, [10] showed that scoring TOEFL essays benefits from features like the number of claims and premises, the number of supported claims, and aspects of tree topology. [20] worked with a broad set of linguistic features and distributions of argument components to predict scores in the ICLE corpus. Closely related to our work is the study by [33] who proposed to use linear "flows" of (coarse) premise and claim units for essay scoring and examined their contribution. We extend this by attending to the more fine-grained features of units.

3 Data

3.1 Argument-Annotated Essays Corpus

We use the AAE corpus [31] as a starting point. The corpus contains 402 student essays annotated for argumentative discourse units (ADU) *major claim*, *claim*, and *premise* and their relations *support* and *attack*. *Major claim* and *claim* are linked via stance annotations. Importantly, components can be extracted from the argumentation structure. Claims always relate to the essay's major claim, while premises support or attack claims (or other premises). Also, while claims and premises can occur in all essay paragraphs, major claims are supposed to be restricted to the first and last paragraphs.

In previous work [26], we annotated the AAE corpus for semantic claim and premise types that can be used for the extraction of argumentative flow patterns. We provided evidence that these flow patterns are suitable for the analysis of argumentation strategy in essays. Here, we will briefly describe the semantic types. For more detailed definitions and examples, we refer the reader to [26]. The following claim types were annotated: *policy*, *value*, and *fact* (see Table 2 below for proportions). *Policy* refers to claims arguing in favor of some action being taken or not being taken. *Value* claims evaluate a target, e.g. they may argue towards it being good/bad or important/unimportant. *Fact*[1] claims,

[1] Note that in this work *fact* does not refer to actual factual statements. Rather it includes claims that the author presents as factual. Determining the actual truth or falsity of a statement, i.e. fact-checking, is beyond the scope of this paper.

on the other hand, state that some target is true or false. In addition to the claim types, we annotated the following premises types: *testimony*, *statistics*, *hypothetical-instance*, *real-example*, and *common-ground*. *Testimony* gives evidence by referring to some expert. *Statistics* uses the results of quantitative research, among others, as evidence. *Hypothetical-instance* and *real-example* are both example categories. The former refers to situations created by the author, i.e. hypothetical situations, while the latter describes actual historical events or a specific statement about the world. Finally, *common-ground* includes common knowledge, self-evident facts, or similar.

In this work, we use the AAE corpus for training ADU identification and semantic type classification models, which are then used to automatically label the Feedback and ICLE corpora with ADUs and their types. Note that we do not use the original relation and stance annotations.

3.2 Feedback Corpus

The Feedback corpus (n = 3,405) is a subset of the PERSUADE corpus [7], which consists of 25,996 essays written by students from grades 6 through 12. In total, 15 prompts were used to elicit the essays. The corpus has been annotated for different ADU types: *lead*, *position*, *claim*, *counterclaim*, *rebuttal*, *evidence*, *concluding statement*. The corpus was additionally annotated for different quality dimensions, such as *cohesion*.

Comparing the argumentative components of the PERSUADE corpus with those of the AAE corpus reveals an apparent overlap in categories. Both corpora are annotated for *claim* and *premise/evidence*. *Position* and *major claim* are defined similarly. However, recall that the ADU types in the AAE corpus are derived from the overall argumentation structure (via the relations between components), while in the PERSUADE corpus, ADUs are defined semantically.

Semantic type classification builds on top of previously classified ADU types. A direct mapping of the ADU types from PERSUADE to AAE would allow us to learn ADU classification on a much larger corpus with more confidence in the predictions for out-of-domain data. To test whether the annotations of the AAE corpus are compatible with those of the PERSUADE corpus, we compare the predictions of our ADU classifier (trained on the AAE data) for the PERSUADE corpus with the original component labels. Mapping the output of our model to the annotations reveals mixed results (see Fig. 1). While *evidence* and *premise* overlap to a good extent, differences in claim conceptualization appear problematic. Both *claim* and *counterclaim* are mapped by similar proportions to *claim* and *premise* by our model. *Rebuttal*, which is defined as "a claim that refutes a counterclaim" [7], is mostly classified as *premise*, while *concluding statement* corresponds to the whole variety of AAE components. Thus, conceptualizations of argument components are on the whole different in the two corpora, and therefore we decided to not use the component annotations of the Feedback corpus, and work with our predicted labels instead.

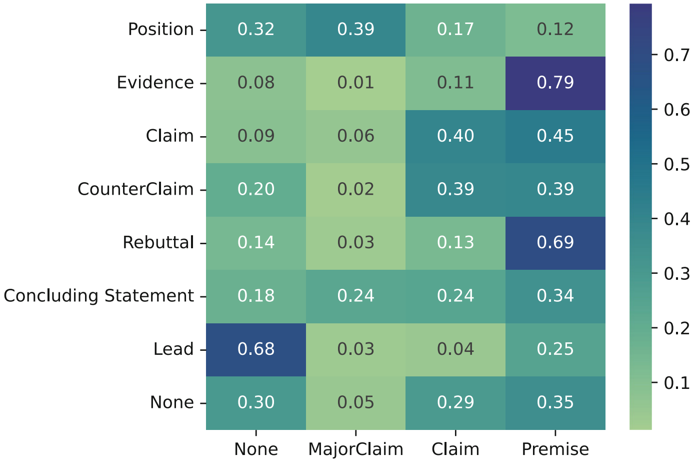


Fig. 1. Confusion matrix for original PERSUADE corpus labels (y-axis) and the predictions of our AAE model (x-axis).

For our quality prediction experiments, we use the dimensions *cohesion* and *conventions*. A text with high *cohesion* is defined as containing a variety of effective linguistic features such as reference and connectives to link ideas across sentences and paragraphs. *Conventions* is defined as the use of common rules, including spelling, capitalization, and punctuation.

3.3 International Corpus of Learner English

Our second target corpus is derived from the ICLE corpus [11], which contains more than 6,000 student essays, of which 91% are argumentative. While no argument component annotations are available, the corpus has been annotated for different scoring dimensions. In this work, we utilize the subset of the corpus that has been annotated for *organization* [19] and *argument strength* [20] (n = 896). Previously a high organization score was defined as *providing a position with respect to an introduced topic and supporting that position* [28]. As this definition roughly describes the core aspects of argumentation, we assume this scoring dimension to be a good candidate for our study. On the other hand, an essay with high *argument strength* "presents a strong argument for its thesis and would convince most readers" [20]. *Argument strength* is thus tied to persuasiveness, again one of the core aspects of successful argumentation.

4 Experiments

Our experiments consist of two steps: Labeling the two target corpora with ADUs and their semantic types (Sect. 4.1), and testing the contribution of type change flows for the task of essay score prediction (Sect. 4.2). In Sect. 4.3, we undertake a qualitative inspection of flows associated with essays of different quality.

4.1 ADU and Sematic Type Classification

We first classify the coarse type of the argumentative components as *major claim*, *claim*, and *premise*. Afterward, we classify the fine-grained semantic types conditioned on their previously identified coarse type. For the semantic type classification, however, we do not distinguish between major claims and claims but regard both of them as *claims*. As both classification tasks, ADU and semantic type, have been studied previously [26,31,33], we do not conduct extensive comparative experiments here but provide the performance of our ensembles for better quality estimation of the projected labels.

We train ensembles of three models each per step. We use 10% of the AAE corpus for development. The remaining data is used for training. Per run, the data is split randomly (with a random number seed set to either 1, 2, or 3).

As a classifier, we use a pre-trained language model, *roberta-base* [17], for both the coarse and the fine-grained step. Following previous work by [33], we identify ADUs solely on the sentence level, disregarding smaller units. Our input to the model is the target sentence plus one additional sentence on the left and the right, to provide context. The context is separated from the target sentence by the model's special tokens. We found that adding this context improves results compared to processing single sentences. Also, it works better than giving the model more context information (additional sentences or structural information such as paragraph breaks).

The ensembles are evaluated on the full AAE corpus. The final classification result is an averaged softmax, from which the label with the maximum probability is chosen. See Table 1 for the results on the annotated corpus. We have further assessed our approach manually on a smaller sample. In particular, we sampled 15 instances per semantic type, and have obtained satisfactory macro results (Claims: 95.55 F1, Premises: 91.64 F1). However, during our review, we noticed some problems with the underlying processed data, e.g. grammatical inconsistencies within sentences and the resulting problems in understanding the author's intentions, which are unfortunately beyond our project's scope.

We then use the trained classification models to predict argument components and semantic types in our two target corpora, Feedback and ICLE. Table 2 shows the distribution of semantic types both for the manually annotated AAE

Table 1. Macro-averaged classification results for the AAE corpus.

Prediction Task	Precision	Recall	F1
ADU Role	82.55	83.24	82.86
Semantic Claim Type	85.23	80.39	81.94
Semantic Premise Type	88.24	69.01	71.30

Table 2. Proportions of semantic types by corpus.

Annotation Class	AAE	Feedback	ICLE
Policy	0.15	0.12	0.17
Value	0.67	0.83	0.79
Fact	0.18	0.05	0.04
Statistics	0.10	0.01	0.03
Hypothetical-Instance	0.24	0.42	0.20
Real-Example	0.19	0.25	0.23
Common-Ground	0.46	0.33	0.53

corpus and for the automatic predictions in the Feedback and ICLE corpora. While some types are equally distributed, e.g. *policy* and *statistics*, there are notable differences in others. For instance, *fact* claims occur more frequently in the AAE essays, while our models labeled claims in Feedback and ICLE more often as *value*. For premises, Feedback contains substantially more *hypothetical-instances*, while the majority class in ICLE is *common-ground*.

4.2 Predicting Essay Quality with Flows of Semantic Types

In this section, we investigate whether essay quality prediction can be improved by using flows of our fine-grained semantic types, in comparison to flows of coarse ADU types, as they had been used by [33]. By "flow", we mean the linear sequence of type labels that occur in a text unit (paragraph or full text). Importantly, we work with *change flows*, which result from collapsing sequences of identical types into a single label. This way, we ignore the information on the "length" of a stretch with the same type and focus only on the changes from one type to another.

To simplify the prediction problem, we group all essays into two classes *good* and *bad*. We normalize all quality scores to the range [0 .. 1], and then label essays with a score above 0.7 as *good* and others as *bad*.

Given the annotations of coarse ADU types and semantic types in the two target corpora, we extract change flow features, both on the global essay level and on that of paragraphs, and for ADU and semantic types, respectively. In Table 3, we show the most common change flows of semantic types in the corpora, divided into first paragraph, body, and last paragraph.

For predicting the quality class, we trained linear models on all extracted change flow features, in particular, we chose stochastic gradient descent models. We set the maximum iteration to 1500, use a balanced class weight, and use grid search cross-validation to decide on the remaining parameters.

Table 3. Most common change flows of semantic types for different argument components. The first letter refers to the type of the argument component (M = major claim, C = claim, and P = premise), the following letters denote the semantic type (e.g. CV = claim-value; PCG = premise-common-ground). Levels are *first* and *last* paragraph of the essay, and everything in-between (*body*).

Level	#	Feedback		ICLE	
		Change Flow	Freq	Change Flow	Freq
first	1	(MV)	13.53%	(PCG)	22.04%
	2	(CV)	9.72%	(MV)	6.82%
	3	(PCG)	4.62%	(PRE)	5.59%
	4	(MV,CV)	4.59%	(CV)	4.92%
	5	(PHI)	3.91%	(PHI)	3.47%
	6	(MP)	2.82%	(PCG,PHI)	3.02%
	7	(PRE)	2.44%	(CV,PCG)	2.91%
	8	(CV,PHI)	2.25%	(PCG,CV)	2.57%
	9	(MV,PHI)	1.76%	(PRE,PCG)	2.35%
	10	(PCG,CV)	1.70%	(PCG,PRE)	2.35%
body	1	(CV)	7.42%	(PCG)	18.81%
	2	(PHI)	7.01%	(CV,PCG)	5.15%
	3	(CV,PHI)	5.93%	(PCG,PHI)	4.20%
	4	(PRE)	4.56%	(PRE)	3.51%
	5	(PCG)	4.36%	(PHI)	3.12%
	6	(CV,PCG)	2.49%	(PCG,PHI,PCG)	2.88%
	7	(PCG,PHI)	2.14%	(CV)	2.80%
	8	(CV,PRE)	2.09%	(PCG,PRE)	2.43%
	9	(CV,PCG,PHI)	2.04%	(PHI,PCG)	2.17%
	10	(CV,PHI,PCG)	1.99%	(PRE,PCG)	2.11%
	11	(PHI,PCG)	1.51%	(PCG,CV)	1.82%
	12	(CV,PHI,CV)	1.28%	(PCG,PRE,PCG)	1.74%
	13	(PHI,CV)	1.09%	(PCG,CV,PCG)	1.40%
	14	(PCG,PHI,PCG)	1.04%	(MV,PCG)	1.08%
	15	(PHI,PCG,PHI)	1.03%	(MV)	1.00%
	16	(CV,PHI,PCG,PHI)	0.99%	(CV,PHI,PCG)	0.98%
	17	(PCG,CV)	0.83%	(CP,PCG)	0.87%
	18	(CV,PRE,PHI)	0.79%	(CV,PHI)	0.77%
	19	(CV,PCG,PHI,PCG)	0.78%	(CV,PCG,PHI)	0.77%
	20	(MV)	0.76%	(CV,PCG,PHI,PCG)	0.77%
last	1	(MV)	13.92%	(MV)	13.87%
	2	(CV)	8.95%	(PCG)	8.61%
	3	(MV,CV)	5.61%	(CV)	5.48%
	4	(MP)	3.27%	(MV,CV)	3.24%
	5	(PHI)	2.15%	(PCG,CV)	3.02%
	6	(CV,MV)	2.08%	(CV,PCG)	2.80%
	7	(PCG)	1.67%	(MP)	2.68%
	8	(MP,CV)	1.60%	(PCG,MV)	2.46%
	9	(CV,PHI)	1.51%	(MV,PCG)	2.46%
	10	(PHI,CV)	1.44%	(PCG,PHI)	2.35%

We run a comparison on 10-fold cross-validation with optimal parameters. Table 4 shows our averaged macro scores (precision, recall, and F1) summarized as mean and standard deviation over 10 runs. We present results for all four essay scoring dimensions *cohesion*, *conventions*, *organization*, and *argument strength*. *Baseline* refers to a stratified classifier, which performs classification based on the observed frequency and outperforms a simple majority voting baseline.

Table 4. Essay Scoring Results. Means and standard deviations of 10-fold cross-validation measured as precision, recall, and F1 scores. As the macro average takes into account the imbalance of the labels, this can result in the F1 values not being between the respective macro values for precision and recall.

Corpus	Dimension	Level	Precision	Recall	F1
Feedback	cohesion	baseline	0.511 (0.032)	0.509 (0.028)	0.509 (0.029)
		adu	0.587 (0.012)	0.631 (0.015)	0.568 (0.027)
		sem	0.575 (0.021)	0.606 (0.033)	0.567 (0.028)
Feedback	conventions	baseline	0.500 (0.022)	0.500 (0.021)	0.499 (0.021)
		adu	0.545 (0.020)	0.573 (0.036)	0.528 (0.026)
		sem	0.566 (0.024)	0.599 (0.039)	0.559 (0.031)
ICLE	organization	baseline	0.506 (0.042)	0.505 (0.040)	0.501 (0.039)
		adu	0.585 (0.055)	0.585 (0.050)	0.580 (0.051)
		sem	0.604 (0.067)	0.607 (0.068)	0.603 (0.068)
ICLE	strength	baseline	0.504 (0.043)	0.503 (0.030)	0.500 (0.036)
		adu	0.522 (0.046)	0.534 (0.065)	0.514 (0.052)
		sem	0.532 (0.056)	0.534 (0.069)	0.519 (0.056)

Both ADU and semantic type models outperform the baselines. We achieve higher F1 scores for the dimensions *conventions* and *organization* with models trained on semantic type change flows instead of coarse ADU type change flows (conventions: 0.559 vs 0.528; organization: 0.603 vs 0.580). For *cohesion* and *argument strength*, the two types of flows obtain similar results.

4.3 Analysis of Feature Impact

We use the trained linear models to extract semantic change flow features that are prevalent in *good* vs *bad* essays and are thus good predictors for the respective class. We normalize the coefficients to center them around zero. Thus, positive coefficients of features in the linear model correlate with yielding a better essay score, while negative coefficients result in worse scores. We investigated the most important change flows in *good* vs *bad* essays both on the full essay level and on the paragraph level. We will only present the results from analyses of the body paragraphs (see Tables 5 and 6), as, presumably, this is where the main argumentation unfolds.

With respect to claim-premise change flows in paragraphs, *bad* essays are more notably characterized by a lack of claims, thus only premises are utilized. This is especially the case for the quality dimensions *cohesion*, *organization*, and *argument strength*. Furthermore, paragraphs of *good* essays appear to show more type variety. This is observable for all quality dimensions, but most clearly for the ICLE corpus, i.e. *organization* and *argument strength*.

More patterns emerge in the premise change flows. For instance, both feedback dimensions (*cohesion* and *conventions*) show the same most dominant flows in good essays, i.e. *PCG-PHI-PCG-PHI* and *PST-PCG*. Also, paragraphs in essays with a high conventions score tend to begin with *common-ground*, while flows exhibit fewer changes than in *bad* essays. Recall that this does not necessarily imply a less complex argumentation structure, as change flows collapse sequences of identical semantic types.

ICLE essays with a high organization score show complex premise change flows, which often include several *common-ground* units framing *hypothetical-instance*, *real-example*, or combinations of those. *Bad* essays, on the other hand, are characterized by example types that are more rarely used in combination with *common-ground*. Similar observations can be made for the *argument strength* dimension. As the feedback corpus, both ICLE dimensions have identical dominant flows, i.e. *PCG-PRE-PCG* and *PCG-PHI-PCG*.

5 Discussion

Transfer across corpora is a complex task. Even corpora that belong to the same general domain of texts, e.g. persuasive essays, may exhibit notable differences in argumentation structure and strategies. This is reflected in the distribution of semantic types across our essay corpora. For instance, the AAE corpus contains a substantially larger proportion of *fact* claims compared to both Feedback and ICLE. The Feedback corpus shows an especially large proportion of *hypothetical-instance*, while premises in the ICLE corpus have been predominantly labeled with *common-ground*. These differences in semantic types have an impact on the observable change flows, and thus on argumentation strategies.

To begin with, *bad* essays with respect to *cohesion*, *organization* and *argument strength* tend to contain paragraphs without a claim more often than *good* essays. This is intuitively plausible, as a full argument typically consists of a claim and at least one premise. However, important change flows for the prediction of bad essays with respect to the *conventions* dimension still contain claims. This may be due to the quality dimension at hand, as *conventions* is less clearly linked to argumentation quality than the other dimensions.

Second, the suitability of premise change flow complexity as a predictor for essay quality depends on the corpus and quality dimension. While ICLE essays with high *organization* and *argument strength* scores tend to show more variety in premise change flow patterns, Feedback essays with high *convention* scores show less variety.

Table 5. Change Flows on Paragraph Level (Body): *Feedback Cohesion & Conventions*. The first letter refers to the type of the argument component (M = major claim, C = claim, and P = premise), the following letters denote the semantic type (e.g. CV = claim-value; PCG = premise-common-ground).

Feedback: Cohesion

	All	Coeff	Claim	Coeff	Premise	Coeff
1	CV-PHI-PRE-PHI	.771	CV	.886	PCG-PHI-PCG-PHI	.966
2	CV-PHI-CV-PCG	.725	CP-CV	.865	PST-PCG	.912
3	CP-PCG-CV	.717	CV-CP	.622	PCG-PHI-PRE-PHI-PRE	.826
4	CV-PCG-CV-PRE-PCG	.702	CV-MV-CV	.621	PCG-PRE-PHI-PRE-PHI	.742
5	PHI-PCG-PHI-PCG	.699	MP	.551	PCG-PHI-PCG-PRE-PHI	.727
6	PHI-CV-PCG	.671	CV-CP-CV	.532	PST-PCG-PHI	.638
7	PRE-PHI-PCG	.658	MV-CP	.445	PCG-PST-PCG	.592
8	CV-PCG-CV-PHI	.656	CF-CV-CF	.426	PCG-PRE-PHI-PRE	.585
9	CV-PHI-PCG-CV	.637	CF-MV	.408	PCG-PRE-PCG-PHI	.545
10	CV-PCG-PHI-PCG	.633	CV-MP	.391	PRE-PCG-PRE-PCG	.521
1	PCG-PHI-PRE-PCG	−.542	MP-CV-MV	−.196	PHI-PCG-PRE-PCG-PHI	−.279
2	PRE-CV	−.543	MP-CF	−.218	PRE-PCG-PHI-PRE-PCG	−.295
3	PCG	−.557	MV-CF	−.248	PRE-PHI-PCG-PHI-PCG	−.299
4	CF	−.572	MF	−.292	PRE-PCG-PHI-PRE	−.302
5	PHI-PRE-PHI	−.574	MV	−.302	PCG-PHI-PRE-PCG	−.318
6	CP-PCG	−.631	MF-CV	−.328	PCG-PHI-PRE-PHI-PCG	−.339
7	MV-PRE	−.665	MP-MV	−.354	PCG-PRE-PCG-PRE	−.349
8	PCG-PHI-PCG-PHI-PCG	−.681	CP-CF	−.399	PRE-PST	−.350
9	PHI-PCG-PRE	−.701	MP-CP	−.512	PCG-PST-PRE-PCG	−.379
10	PCG-PHI-PRE	−.728	CF-CV	−.957	PCG-PHI-PCG-PHI-PCG-PHI-PCG	−.545

Feedback: Conventions

	All	Coeff	Claim	Coeff	Premise	Coeff
1	PRE-MV-PRE	.783	MP-CV	.876	PCG-PHI-PCG-PHI	.869
2	MV-PCG-PRE	.774	CV	.452	PST-PCG	.798
3	CV-PCG-PHI-CV-PCG	.693	CP-CV	.407	PCG-PRE-PHI-PRE-PCG	.743
4	PHI-PCG	.690	MV-CP-CV	.378	PCG-PST-PRE	.692
5	CV-PRE-CV	.687	CV-CF-CV	.364	PCG-PST-PCG	.666
6	CV-PHI-PRE-PHI	.679	MP-CF	.344	PCG-PHI-PRE	.661
7	PCG-PRE-PCG-PHI-CV	.664	CP-MV	.339	PCG-PST-PHI	.636
8	CF-PRE-CV	.654	CV-CP-CV	.274	PRE-PCG-PRE-PCG-PRE	.635
9	PCG-PHI-PCG-CV	.632	CV-CP-MV	.273	PST-PHI	.615
10	CV-PRE-PHI-MV	.601	CP-MP	.257	PHI-PST	.612
1	CV-PST	−.569	CV-CF	−.141	PHI-PST-PHI-PCG	−.301
2	CF	−.574	CF-CV	−.147	PCG-PHI-PCG-PRE	−.308
3	CV-PCG-CV-PCG	−.592	MP-MV	−.152	PHI-PST-PCG	−.337
4	MV-PRE	−.615	MV-CV-CP	−.199	PRE-PHI-PCG-PRE	−.348
5	PHI-PRE-CV	−.628	CF-CP	−.248	PRE-PCG-PHI-PCG-PHI	−.378
6	CF-PHI	−.646	CV-MV	−.306	PRE-PST	−.401
7	CV-PRE-PCG	−.647	MV-CF	−.306	PHI-PCG-PHI-PCG-PRE	−.450
8	CV-CP	−.750	MF-CV	−.339	PRE-PCG-PRE-PHI	−.530
9	CV-PCG-CV	−.777	CV-MF	−.346	PCG-PHI-PRE-PCG	−.587
10	PCG-CV-PCG	−.783	CV-MP-CV	−.513	PHI-PCG-PRE-PHI	−.611

Table 6. Change Flows on Paragraph Level (Body): *ICLE Organization* & *Argument Strength*. The first letter refers to the type of the argument component (M = major claim, C = claim, and P = premise), and the following letters denote the semantic type (e.g. CV = claim-value; PCG = premise-common-ground).

ICLE: Organization

	All	Coeff	Claim	Coeff	Premise	Coeff
1	CV-PCG-PHI	.010	CP-CV	.008	PCG-PRE-PCG	.016
2	PCG-PHI-PCG	.008	CV	.007	PCG-PHI-PCG	.013
3	PCG-PRE-PCG	.007	CV-CP	.003	PHI-PCG	.010
4	CV-PHI-PCG	.005	CV-CF	.003	PCG	.008
5	CV-PCG	.005	MP-CP	.001	PHI-PCG-PHI-PCG	.005
6	PCG-PHI-PRE-PCG	.004	CV-CF-CP	.001	PST-PCG	.004
7	CV-PCG-PRE-PCG	.004	MF	.001	PCG-PST-PCG	.003
8	CV-PCG-PHI-PCG	.003	CF-CP	.001	PCG-PHI-PRE-PCG	.003
9	PHI-PCG-PHI-PCG	.003	MP-CV	.000	PCG-PHI-PCG-PHI-PCG-PHI	.003
10	PCG-CV-PCG-PHI-PCG	.003	CV-CP-CV	.000	PCG-PRE-PCG-PRE-PCG	.003
1	PRE-PCG-PRE	−.005	CV-CF-CV	−.001	PRE-PHI-PRE	−.002
2	MV	−.006	CV-MV	−.001	PRE-PHI	−.003
3	CP	−.006	CV-MP	−.001	PHI-PCG-PHI	−.004
4	PRE-PCG	−.006	MP	−.001	PHI-PRE	−.006
5	PCG-PRE	−.008	CF-CV	−.001	PRE-PCG-PRE	−.006
6	PRE	−.008	MV-CP	−.003	PCG-PHI	−.007
7	PCG-PHI	−.013	MV-CV	−.004	PCG-PRE	−.009
8	CV	−.014	CF	−.006	PHI	−.011
9	PHI	−.014	MV	−.011	PRE	−.015
10	PCG	−.022	CP	−.015	PRE-PCG	−.015

ICLE: Argument Strength

	All	Coeff	Claim	Coeff	Premise	Coeff
1	CV-PCG-PHI-PCG	.013	CV	.009	PCG-PRE-PCG	.012
2	CV-PCG	.012	MP	.005	PCG-PHI-PCG	.011
3	PCG-PRE-PCG	.006	CP	.004	PCG-PHI-PCG-PRE-PCG	.006
4	PCG-PHI-CV	.005	CV-MV	.003	PCG-PST-PCG	.006
5	CV-PHI-PCG-PHI	.005	CP-CV	.002	PCG-PHI-PCG-PHI-PCG	.005
6	CP-PHI	.004	CV-CF-CV	.002	PHI-PRE-PCG	.005
7	CP	.004	MF	.002	PHI-PCG-PHI-PCG	.005
8	PRE-PHI-CV	.004	CV-CF-CP	.001	PRE-PHI-PCG	.004
9	CV-PHI-PCG-PHI-PCG	.003	MP-CP	.001	PCG-PHI-PST	.003
10	PCG-PST	.003	CV-MV-CV	.001	PST-PCG-PHI-PCG	.003
1	CV-PCG-PRE	−.004	CF-CP	.001	PHI-PCG	−.003
2	PRE-PHI	−.004	CF-CV	.000	PCG-PHI	−.003
3	PRE	−.005	MP-CV	−.001	PRE-PCG-PRE-PCG	−.003
4	PCG-CV	−.005	CV-CP-CV	-.001	PCG-PRE-PHI	−.004
5	CV-PHI-PCG	−.007	CV-CP	-.001	PHI-PRE	−.005
6	MV	−.007	MV-CP	−.002	PRE-PCG-PRE	−.005
7	CV	−.010	CV-CF	−.003	PST-PCG	−.005
8	PHI	−.010	MV-CV	−.005	PHI	−.007
9	PCG	−.012	CF	−.006	PRE	−.008
10	PCG-PHI	−.016	MV	−.022	PRE-PCG	−.013

Third, *good* essays with respect to *conventions*, *organization*, and *argument strength* show change flows that begin with *common-ground* or use it as a framing type, typically in combination with an example type. This is in line with the argumentation strategy found in the AAE corpus of beginning (and ending) an argument with a general observation while inserting more concrete premises, e.g. examples, in between [26]. Overall, we can summarize that semantic change flows can be indicative of argument strategies applied to produce a persuasive essay of high quality.

6 Conclusion

In this work, we studied the question to what extent argument arrangement in the sense of change flows of semantic types can support the prediction of student essay quality.

To this end, we trained models for ADU and semantic type classification on the AAE corpus, which has been annotated accordingly in previous work [26,31]. We used these models to label essays in two target corpora: Feedback and ICLE. We extracted change flows of ADUs and semantic types and used them for essay quality prediction. Importantly, we showed that some dimensions of essay quality, i.e. *conventions* and *organization*, can be predicted better by using flows of semantic types rather than by coarse ADU types. This result expands on the earlier work of [33]. Finally, we identify change flow features that are important predictors for *good* vs *bad* essays.

We find that 1) the distribution of semantic types depends on the corpus at hand and 2) *bad* essays tend to lack claims, i.e. contain incomplete arguments. Further, we observe that 3) the mere complexity of change flows is not a sufficient predictor for quality and 4) certain change flows of semantic types indicate the use of argumentation strategies.

In the future, we are interested in investigating more thoroughly the relationship between argumentation strategies and essay quality. Here, we considered this topic only briefly in Sect. 5. Also, we plan to extend our analysis to other out-of-domain corpora (e.g., news editorials and the subreddit Change My View).

Limitations

Due to the very small number of annotated essays (402 instances), it is only possible to estimate to a limited extent how the projection of the annotations by our neural models onto corpora outside the essay domain works. The questions of how well these models work on out-of-domain data and how well the semantic type scheme applies to other domains deserve greater attention in future work.

For our study, we decided to follow previous research that simplifies the argument component classification to the sentence level. Although this is considered legitimate for the AAE corpus due to the consistently strict essay structure, in general, this is a simplification that leads to inexactness in the extracted components.

Our work is the first attempt to use abstract semantic patterns to measure the quality of student writing. However, due to the relatively small gains in performance, we assume that the selected quality dimensions may not ideally capture the meaning of our semantic types.

Acknowledgement. This research has been supported by the German Research Foundation (DFG) with grant number 455911521, project "LARGA" in SPP "RATIO".

References

1. Addawood, A., Bashir, M.: "what is your evidence?" A study of controversial topics on social media. In: Proceedings of the Third Workshop on Argument Mining, Berlin, Germany, pp. 1–11. ACL (2016)
2. Al-Khatib, K., Wachsmuth, H., Kiesel, J., Hagen, M., Stein, B.: A news editorial corpus for mining argumentation strategies. In: Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers, Osaka, Japan, pp. 3433–3443. The COLING 2016 Organizing Committee (2016)
3. Beigman Klebanov, B., Stab, C., Burstein, J., Song, Y., Gyawali, B., Gurevych, I.: Argumentation: content, structure, and relationship with essay quality. In: Proceedings of the Third Workshop on Argument Mining (ArgMining2016), Berlin, Germany, pp. 70–75. ACL (2016)
4. Carlile, W., Gurrapadi, N., Ke, Z., Ng, V.: Give me more feedback: annotating argument persuasiveness and related attributes in student essays. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, pp. 621–631. ACL (2018)
5. Chen, W.F., Chen, M.H., Mudgal, G., Wachsmuth, H.: Analyzing culture-specific argument structures in learner essays. In: Proceedings of the 9th Workshop on Argument Mining, pp. 51–61. International Conference on Computational Linguistics, Online and in Gyeongju, Republic of Korea (2022)
6. Chen, Z., Verdi do Amarante, D., Donaldson, J., Jo, Y., Park, J.: Argument mining for review helpfulness prediction. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, Abu Dhabi, United Arab Emirates, pp. 8914–8922. ACL (2022)
7. Crossley, S.A., Baffour, P., Tian, Y., Picou, A., Benner, M., Boser, U.: The persuasive essays for rating, selecting, and understanding argumentative and discourse elements (persuade) corpus 1.0. Assessing Writing **54**, 100667 (2022)
8. Daxenberger, J., Eger, S., Habernal, I., Stab, C., Gurevych, I.: What is the essence of a claim? Cross-domain claim identification. In: Palmer, M., Hwa, R., Riedel, S. (eds.) Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, pp. 2055–2066. ACL (2017)
9. Dusmanu, M., Cabrio, E., Villata, S.: Argument mining on Twitter: arguments, facts and sources. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, pp. 2317–2322. ACL (2017)
10. Ghosh, D., Khanam, A., Han, Y., Muresan, S.: Coarse-grained argumentation features for scoring persuasive essays. In: Erk, K., Smith, N.A. (eds.) Proceedings of the 54th Annual Meeting of the ACL (Volume 2: Short Papers), Berlin, Germany, pp. 549–554 (2016). https://doi.org/10.18653/v1/P16-2089

11. Granger, S., Dagneaux, E., Meunier, F., Paquot, M., et al.: International corpus of learner English, vol. 2. Presses universitaires de Louvain Louvain-la-Neuve (2009)
12. Granger, S., Dupont, M., Meunier, F., Naets, H., Paquot, M.: International Corpus of Learner English. Version 3. Presses universitaires de Louvain (2020)
13. Hidey, C., Musi, E., Hwang, A., Muresan, S., McKeown, K.: Analyzing the semantic types of claims and premises in an online persuasive forum. In: Proceedings of the 4th Workshop on Argument Mining, Copenhagen, Denmark, pp. 11–21. ACL (2017)
14. Higgins, C., Walker, R.: Ethos, logos, pathos: strategies of persuasion in social/environmental reports. Account. Forum **36**(3), 194–208 (2012). Analyzing the Quality, Meaning and Accountability of Organizational Communication
15. Hua, X., Wang, L.: Understanding and detecting supporting arguments of diverse types. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, Canada, pp. 203–208. ACL (2017)
16. Lawrence, J., Reed, C.: Argument mining: a survey. Comput. Linguist. **45**(4), 765–818 (2020)
17. Liu, Y., et al.: Roberta: a robustly optimized BERT pretraining approach. CoRR abs/1907.11692 (2019)
18. Ong, N., Litman, D., Brusilovsky, A.: Ontology-based argument mining and automatic essay scoring. In: Proceedings of the First Workshop on Argumentation Mining, Baltimore, Maryland, pp. 24–28. ACL (2014)
19. Persing, I., Davis, A., Ng, V.: Modeling organization in student essays. In: Li, H., Màrquez, L. (eds.) Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Cambridge, MA, pp. 229–239. ACL (2010)
20. Persing, I., Ng, V.: Modeling argument strength in student essays. In: Zong, C., Strube, M. (eds.) Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Beijing, China, pp. 543–552. ACL (2015). https://aclanthology.org/P15-1053
21. Persing, I., Ng, V.: End-to-end argumentation mining in student essays. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, California, pp. 1384–1394. ACL (2016)
22. Persing, I., Ng, V.: Unsupervised argumentation mining in student essays. In: Proceedings of the Twelfth Language Resources and Evaluation Conference, Marseille, France, pp. 6795–6803. ELRA (2020)
23. Rinott, R., Dankin, L., Alzate Perez, C., Khapra, M.M., Aharoni, E., Slonim, N.: Show me your evidence - an automatic method for context dependent evidence detection. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, pp. 440–450. ACL (2015)
24. Sazid, M.T., Mercer, R.E.: A unified representation and a decoupled deep learning architecture for argumentation mining of students' persuasive essays. In: Proceedings of the 9th Workshop on Argument Mining, pp. 74–83. International Conference on Computational Linguistics, Online and in Gyeongju, Republic of Korea (2022)
25. Schaefer, R., Knaebel, R., Stede, M.: On selecting training corpora for cross-domain claim detection. In: Lapesa, G., Schneider, J., Jo, Y., Saha, S. (eds.) Proceedings of the 9th Workshop on Argument Mining, pp. 181–186. International Conference on Computational Linguistics, Online and in Gyeongju, Republic of Korea (2022)

26. Schaefer, R., Knaebel, R., Stede, M.: Towards fine-grained argumentation strategy analysis in persuasive essays. In: Alshomary, M., Chen, C.C., Muresan, S., Park, J., Romberg, J. (eds.) Proceedings of the 10th Workshop on Argument Mining, Singapore, pp. 76–88. ACL (2023)
27. Schaefer, R., Stede, M.: GerCCT: an annotated corpus for mining arguments in German tweets on climate change. In: Proceedings of the Thirteenth Language Resources and Evaluation Conference, Marseille, France, pp. 6121–6130. ELRA (2022)
28. Silva, T.: Toward an understanding of the distinct nature of L2 writing: the ESL research and its implications. TESOL Q. **27**(4), 657–677 (1993)
29. Song, Y., Heilman, M., Beigman Klebanov, B., Deane, P.: Applying argumentation schemes for essay scoring. In: Proceedings of the First Workshop on Argumentation Mining, Baltimore, Maryland, pp. 69–78 (2014). https://doi.org/10.3115/v1/W14-2110
30. Stab, C., Gurevych, I.: Identifying argumentative discourse structures in persuasive essays. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Doha, Qatar, pp. 46–56. ACL (2014)
31. Stab, C., Gurevych, I.: Parsing argumentation structures in persuasive essays. Comput. Linguist. **43**(3), 619–659 (2017)
32. Stede, M., Schneider, J.: Argumentation Mining. Synthesis Lectures in Human Language Technology, vol. 40. Morgan & Claypool (2018)
33. Wachsmuth, H., Al-Khatib, K., Stein, B.: Using argument mining to assess the argumentation quality of essays. In: Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers, pp. 1680–1691. The COLING 2016 Organizing Committee, Osaka, Japan (2016)
34. Wang, H., Huang, Z., Dou, Y., Hong, Y.: Argumentation mining on essays at multi scales. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 5480–5493. International Committee on Computational Linguistics, Barcelona, Spain (Online) (2020)

Finding Argument Fragments on Social Media with Corpus Queries and LLMs

Nathan Dykes[1](✉), Stephanie Evert[1], Philipp Heinrich[1], Merlin Humml[2], and Lutz Schröder[2]

[1] Chair of Computational Corpus Linguistics, Friedrich-Alexander-Universität Erlangen-Nürnberg, Bismarckstr. 6, 91054 Erlangen, Germany
{nathan.dykes,stephanie.evert,philipp.heinrich}@fau.de

[2] Chair of Theoretical Computer Science, Friedrich-Alexander-Universität Erlangen-Nürnberg, Martensstr. 3, 91058 Erlangen, Germany
{merlin.humml,lutz.schroder}@fau.de

Abstract. We are concerned with extracting argumentative fragments from social media, exemplified with a case study on a large corpus of English tweets about the UK Brexit referendum in 2016. Our overall approach is to parse the corpus using dedicated corpus queries that fill designated slots in predefined logical patterns. We present an inventory of logical patterns and corresponding queries, which have been carefully designed and refined. While a gold standard of substantial size is difficult to obtain by manual annotation, our queries can retrieve hundreds of thousands of examples with high precision. We show how queries can be combined to extract complex nested statements relevant to argumentation. We also show how to proceed for applications needing higher recall: high-precision query matches can be used as training data for an LLM classifier, and the trade-off between precision and recall can be freely adjusted with its cutoff threshold.

Keywords: Argument extraction · Semantic parsing · Corpus queries · Social media · LLMs

1 Introduction

We report on a methodology for extracting arguments from posts in social media, with the overarching aim of gaining an overview of arguments and views being voiced. Argumentation on social media is characterized by a high degree of informality, which makes our endeavour, viz. mapping the argumentative landscape on social media, particularly difficult.

In our central case study, we analyse English tweets about the Brexit referendum in 2016. Tweets are particularly challenging because they are too short to contain fully structured arguments (i.e., premises, conclusions, and links relating the argument parts), see e.g. Bhatti, Ahmad, and Park [3]. Therefore, we aim for a semantically precise method to capture argument fragments by parsing

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 163–181, 2024.
https://doi.org/10.1007/978-3-031-63536-6_10

them into a pre-defined but extensible set of logical patterns, i.e. formulae with placeholders in dedicated modal logics. For each logical pattern, we formulate multiple dedicated corpus queries [11] reflecting different linguistic realizations of the same type of statement. It is expected that such an approach will have comparatively low recall but (usually) very high precision. This in fact suits our intention to map general tendencies in the argumentative landscape and reconstruct complete arguments from precisely parsed fragments, rather than extracting each individual instance of an argument type.

In this paper, we demonstrate our overall workflow, query development and evaluation results; in particular confirming that we arrive at high precision through corpus queries. In a subsequent extension of the approach, we experiment with *hierarchical* patterns and associated queries, where we look for matches of an inner pattern in the text spans corresponding to placeholders of an outer pattern. Moreover, we present an approach that uses query results as training data for an LLM classifier in order to change the precision-recall trade-off, with promising results.

Related work Work on argument mining in social media often focuses on graphical structure (e.g. [1,15]; see also Lytos et al. [19] for a survey, and Lytos et al. [20] for an example of a recent, purely data-driven, approach), and has highlighted linguistic and logical challenges [4,7,14].

Our high-precision approach based on logical patterns and queries appears to be new as such, and is distinct in particular from text mining with knowledge patterns (e.g. [6,23]). Work on the extraction of counterfactuals [32] follows partly similar methods, but uses regular expressions instead of linguistically informed corpus queries.

Recent work on argument mining in Twitter has concentrated on identifying high-level categories such as *argumentative* vs. *non-argumentation*, *factual* vs. *opinion*, *claim* vs. *support* vs. *rebuttal*, etc., which specify the general role of each tweet in an argument (e.g. [2,12,30]). This is much more coarse-grained than (and also fundamentally different from) our approach of extracting the content of argumentative fragments in the form of logical patterns.

NLP approaches to argument mining often focus on automatic classification of such categories by training machine learning algorithms (e.g. a support vector machine or logistic regression), see e.g. [5] for a survey. However, with only a handful of positive examples every one hundred tweets (see Sect. 3.2), obtaining a sufficient amount of training examples is prohibitively expensive. Recent work on large language models (LLMs), on the other hand, promises to leverage linguistic knowledge derived from unlabeled data; these models only have to be *fine-tuned* on the classification task at hand [see e.g. 25,26]. We will show in this paper that fine-tuning a general-purpose LLM on a handful of positive examples does not yield satisfying results. A more competitive approach are frameworks for few-shot learning. In SetFit [33], a pre-trained Sentence Transformers [27] model is first fine-tuned on a number of contrastive pairs of labelled texts and then used to encode the training data. Finally, a text classification head is trained using the encoded data.[1]

Our combined approach outlined below, i.e. leveraging corpus queries for training an LLM, is in fact similar to *data augmentation* [13], i.e. increasing the diversity of training examples without collecting new data. However, data augmentation usually works by creating artifical examples that are very similar to the original positive examples (or are made up altogether), while our approach uses only authentic examples as training data. Finally, in the case that high-quality training data are not available, one could use "weak labeled data" [31], applying coarse heuristics to extract training examples while allowing for a significant amount of noise. This also bears similarities to our approach, but is fundamentally different from our high-precision, low-recall strategy.

2 Argumentative Fragments

Given the complexity of natural language argumentation, we approach argument mining in a piece-by-piece manner, aiming to parse argument fragments from our inventory of predefined logical patterns by means of high-precision corpus queries. This means we define logical patterns expressing forms of propositions used in arguments which then have multiple corresponding corpus queries each covering multiple syntactic realizations to express such a proposition.

2.1 An Inventory of Logical Patterns

Starting from an analysis of argument schemes in the style of Reed, and Macagno [34], we created an initial inventory of logical patterns for argumentation. This inventory was extended with additional patterns that were common in our data but outside of the standard catalogue of argumentation schemes, to adapt to the informality of arguments on social media.

The patterns are formulae with sorted placeholders in dedicated modal logics. A typical example is the *desire* pattern $D_{\{?0:entity\}}\{?1 : formula\}$ expressing that entity ?0 wants formula ?1 to become true. Note that the sort *entity* does not require the expression to evaluate to a single entity but instead describes an abstract group of entities in the sense of Humml and Schröder [17]. Patterns can also go beyond single modality statements to more complicated formulae or even sets of formulae (or equivalently conjunctions) like, e.g., the *group knowledge* pattern $K_{\{?0:entity\}}(\{?1 : formula\}); (?2) \implies (?0)$ expressing that entity ?2 is part of entity ?0 whose members know that formula ?1 is true. The underlying semantics of abstract group knowledge then implies that ?2 also knows ?1, i.e. $K_{\{?2:entity\}}(?1)$. This pattern could, e.g., be an indicator for an argument from *Position to Know* [34]. The logical framework we use has been described in more detail in earlier work [8,9]. As indicated above, the overall character of the

[1] Other state-of-the-art methods such as T-Few [18] might yield even better results, but SetFit is a convenient and widely-used framework that does not require any prompt-engineering.

representation logic is *modal* in the sense that it features operators expressing that statements hold in a certain way; e.g. the operators D and K ('desires' / 'knows') featuring in the above examples are modal operators.

Our motivation for extracting argumentative content in the form of logical statements is to leverage automated reasoners to aid in reconstructing complete arguments. In everyday argumentation, it is uncommon and even socially unacceptable to give detailed arguments that mention every reasoning step and every premise. Instead, dialogical argumentation relies on shared common knowledge between the dialogue participants to complete the missing parts of arguments. In our processing pipeline, the logical reasoner and a knowledge base are eventually intended to take the place of the human reasoner trying to fill in the missing pieces of the arguments. For example the reasoner might combine desire statements into larger desired states of the world. In our corpus of tweets the queries retrieve two desire statements attributed to Cameron: "PROOF Cameron WANTS Turkey to join the EU [...]" "Ersatz 'reform deal' proves Cameron always wanted the UK to stay in the EU [...]" The reasoner would then draw the conclusion that Cameron wants both the UK and Turkey to be in the EU.

2.2 Nested Patterns

Combining patterns from our inventory can yield many variants of more complex statements. We follow a recursive approach to extracting relevant information from selected pattern combinations. Empirically, we apply corpus queries to the text spans matching placeholders of an "outer" pattern in order to find matches of further "inner" patterns. The sorting discipline on placeholders then implies a corresponding sorting discipline on the patterns themselves with different logic syntaxes used in patterns of different sorts. For example, a formula describing an entity will employ different modalities than a formula defining an action or a truth statement. In Fig. 1, the entity slot in the *desire* pattern is expanded by filling in a more complicated entity expression from the set of entity patterns; in the example, this expression denotes the intersection of two entities (remember that entities are abstract groups). Similarly, the placeholder ?1, which represents a truth statement, could be expanded with a more concrete pattern, such as *membership* [8]. A concrete realization of the doubly extended pattern would be an expression like "trustworthy economists favour the UK being in the EU", which

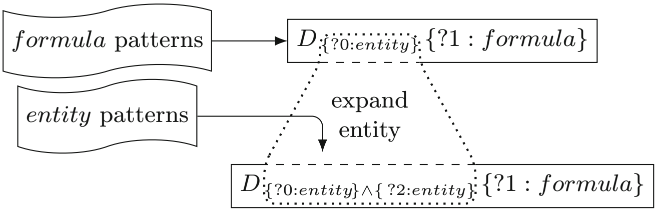

formula patterns
entity patterns
$D_{\{?0:entity\}}\{?1 : formula\}$
expand entity
$D_{\{?0:entity\}\wedge\{?2:entity\}}\{?1 : formula\}$

Fig. 1. A nested pattern

would be expressed as the formula $D_{\mathsf{isTrustworthy} \wedge \mathsf{isEconomist}}(\mathsf{UK} \rightarrow \mathsf{EU})$. In practice, this approach is implemented by hierarchical queries as discussed in Sect. 5.

3 Data

3.1 Corpus and Linguistic Annotation

We use the data set collected by Dykes et al. [9], which consists of tweets containing the token *brexit* (case-insensitive, with or without hashtag marker) collected before and after the UK Brexit referendum in 2016. Our final corpus only includes original tweets and has been filtered with a deduplication algorithm (which disregards case differences, @-mentions, URLs, and hashtags). It amounts to ca. 4.3 million tweets with a total size of ca. 80 million tokens, most of them posted close to the actual referendum on June 23, 2016.

The linguistic annotation pipeline follows [8], comprising Ark TweetNLP [22][2] for simple PoS tags, the OSU Twitter NLP tools [28][3] for Penn-style PoS tags and named entity recognition, and a lemmatizer based on Minnen, Carroll, and Pearce [21]. Sentence boundaries are inserted using SoMaJo [24][4]. Note that these systems generate different tokenization layers, which are reconciled in post-processing. The final corpus is indexed using the IMS Open Corpus Workbench [10].[5]

3.2 Manual Annotation of Argument Fragments

Several student assistants were hired to annotate relevant argument fragments in two random corpus samples: `pre` contains 785 tweets from before the referendum, `post` contains 973 tweets from August 21, 2016 (i.e. after the referendum). Doubtful cases were discussed with the project members, and all disagreements in annotation were adjudicated on a regular basis. Note that prevalence of patterns is low: only about 5–7% of all tweets contain *desire* and 7–10% contain *membership*, cf. Table 2. Additionally, random samples of query matches (data sets `matches`) were annotated, amounting to a total of 3997 tweets for the *desire* pattern and 1005 tweets for *membership*.

Annotation was highly time-consuming, thus it was unrealistic to obtain a sufficient number of examples to train a machine-learning classifier to detect patterns automatically. Several factors contributed to this challenge: Firstly, the aforementioned low prevalence of patterns meant that annotators needed to check vast numbers of tweets. Secondly, the linguistic realizations that do occur take many different forms even within the same pattern, which made it easy to miss relevant cases. Thirdly, despite working with detailed annotation guidelines, the decision of whether a given expression fits a particular logical formula or not

[2] http://www.cs.cmu.edu/~ark/TweetNLP/.

[3] https://github.com/aritter/twitter_nlp.

[4] https://github.com/tsproisl/SoMaJo.

[5] https://cwb.sourceforge.io/.

still proved difficult. These difficulties are reflected in the (sometimes low) inter-annotator agreement scores in Table 1. It is worth noting that the annotators where instructed to err on the side of annotating doubtful cases positively, with corner cases included in the subsequent adjudication process.

Table 1. Kappa scores for the three most annotated patterns. *E*, *M*, and *V* represent three independent student assistants, *gold* was obtained in a subsequent adjudication process.

(a) *desire*

	E	M	V
M	0.39	-	-
V	0.79	0.39	-
gold	0.46	0.86	0.45

(b) *membership*

	E	M	V
M	0.66	-	-
V	0.59	0.62	-
gold	0.80	0.72	0.63

(c) *opposition*

	E	M	V
M	0.35	-	-
V	0.59	0.42	-
gold	0.36	0.76	0.46

4 Corpus Queries

4.1 Methods

The manually annotated instances of our argument patterns serve as templates for specialized corpus queries. For the case of *desire*, recall that our aim is to find realisations of the formula $D_{\{?0:entity\}}\{?1 : formula\}$. The following statements are examples of posts expressing a *desire* according to our guidelines:

1. "without giving u reasons for u to argue with, I think *I'm in favour of an exit*!!"
2. "*Several key @vote_leave folks on record wanting to privatise #NHS* & #Brexit #Tory ministers never showed any concern for NHS @stariep"
3. "@SadiqKhan Sir, are *you in favor of #Brexit*?"
4. "eAndrew Neil is chair of *@spectator which has come out for #Brexit* How can @afneil still be allowed control of #BBCSDP #BBCDP?"
5. "*Bryan Adams is in favour of Brexit.*"

While these examples all correspond to our formula, they are clearly not identical linguistically. Variation occurs in terms of what the entity and the formula refer to, as well as how each concept is expressed regarding lexis and syntax. Nevertheless, examples 1, 3 and 5 can be generalised on the linguistic level to *ENTITY is in favour of FORMULA*. Based on such similarities, we constructed the following query to extract further similar instances of *desire* from the overall corpus:

```
@0[::] /entity_np_actor[] @1[::]
[xpos=‘‘MD’’ | lemma=‘‘be|have’’ | upos=‘‘ADJ|ADV’’]*
[lemma=‘‘in’’] [upos=‘‘ADJ’’]* [lemma=$nouns_desire]
[lemma=‘‘for|of|pro|that|to’’]
@2[::] (/entity_np_all[] | [xpos=‘‘VBG’’]) (/lexical_words[])
    ↪ * @3[::]
```

Our queries are written in the query language [11] of the corpus query processor (CQP) of the IMS Open Corpus Workbench (CWB, [10]), enabling efficient execution of complex queries in large corpora. The query language is based on regular expressions over token descriptions, which are Boolean expressions of attribute-value pairs (where values can again be matched by regular expressions). For example, `[lemma=‘‘be’’]` retrieves all forms of *BE* (*be, am, are, is, was, were, been, being*), and `[upos=‘‘ADJ’’]*` retrieves sequences of adjectives. Additionally, structural annotation elements (such as tweets, paragraphs or sentences) can be matched by XML tags, e.g. `<tweet> []* </tweet>` for a complete tweet.

The most important parts of this query are the slot fillers. For *desire*, they represent the ENTITY and FORMULA slots. In the corresponding query, the tokens belonging to a given slot are enclosed in target markers: `@0[::] ...` ↪ `@1[::]` (ENTITY) and `@2[::] ... @3[::]` (FORMULA). The ENTITY is modelled with a CQP macro `/entity_np_actor[]`, which expands to match noun phrases containing personal pronouns, proper names, or nouns from a word list referencing people or organizations (e.g. *politician* or *party*). Limiting the noun phrase in this way ensures that the expression in the ENTITY slot can reasonably be expected to express a meaningful desire. While we will necessarily lose some recall with this restriction, a more flexible ENTITY slot filler would compromise precision too much. However, the word lists were extended using word embeddings, which we used to suggest distributionally similar items to the ones that had been collected manually. The macro `/entity_np_all[]` in the FORMULA region matches a much more general noun phrase, since FORMULA can refer to a wider range of concepts. Alternatively, this slot can be realized with a verb in gerund form (e.g. *to be in favour of exiting*), followed by an arbitrary number of content words within the same tweet. The middle part of the query provides linguistic structure to ensure that it actually matches an expression of *desire*. After optional modifiers, its main part is *in favour/hope/support/... for/of/...*.

Development Environment. There are two major shortcomings to the main CWB user interface CQPweb [16], which we initially used for query development (similar limitations apply to other tools like AntConcc[6] or Sketch Engine[7]). Firstly, when writing a large repertoire of queries, reusable elements like word lists and macros need to be managed efficiently. While CWB can easily read

[6] https://www.laurenceanthony.net/software.

[7] https://www.sketchengine.eu/.

macros and word lists from plain text files, CQPweb does not provide access to these files.

More importantly, these tools are designed with traditional corpus studies in mind, which typically use much shorter queries. Accordingly, functionality for displaying and sorting query results is usually optimised for single words and short expressions. In our usage scenario, i.e. argument queries, it is crucial to mark and highlight multiple positions within query matches. Queries that retrieve surface realizations of the *desire* pattern need to specify two *slots* (text spans) representing the ENTITY and FORMULA placeholders, respectively.

It has only recently become possible in version 3.4.16 of CWB to mark more than a single position inside a query match (in addition to start and end of the match), using anchors `@0`, …, `@9`. This new feature requires support from a wrapper application, though, which has to run every query up to 5 times, collecting two anchor positions in each step. We provide the Python library *cwb-ccc*[8], which includes such a wrapper.[9] Anchor positions can also be adjusted by an integer offset; this is especially helpful if the query contains optional elements (with quantifiers `?`, `+` or `*`).

Since developing corpus queries is an iterative hermeneutic process, carefully balancing precision and recall for the task at hand, it would be very inconvenient to run a wrapper from the command line and collect its results whenever a query is modified. We thus developed *Spheroscope*[10], a web app specifically dedicated to the iterative development of corpus queries.

Here, queries can use an arbitrary number of word lists and macros, which can be stored and re-used via the user interface. The interface also enables users to obtain the frequencies of all words from a word list for any given corpus. Additionally, semantically similar words can be suggested for semi-automatic extension of word lists. For semantic similarity, we use custom word embeddings trained on a larger, independent sample of English tweets. Similar items can be sorted by their corpus frequency or by cosine similarity (by default, up to 200 items are displayed, hapax legomena excluded). Similarly, macros can be defined, named, stored, inspected (frequency breakdown), and reused via the user interface.

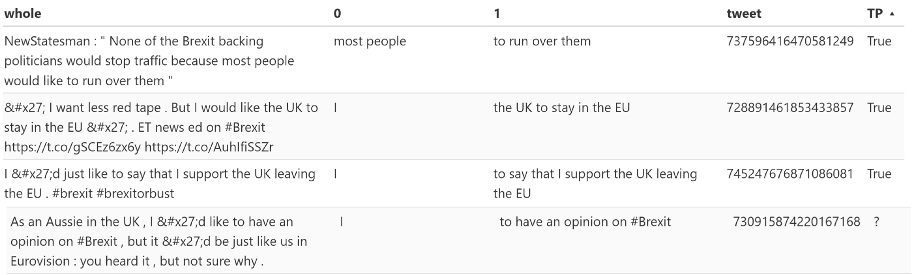
Iterative Query Development. In order to incorporate feedback from manual annotation and to reflect our developing understanding of possible realizations of our continuously refined inventory of argument fragments, query development is necessarily iterative. This affects evaluation, since precision and recall need to be reassessed with every change to the queries. Recall can only be measured on random subsets of tweets; precision can be assessed qualitatively by reading concordance lines of query matches as well as quantitatively by using labelled examples. The development environment thus directly indicates for each query

[8] https://pypi.org/project/cwb-ccc/.

[9] The module provides additional functionality for tradtional corpus linguistic tasks such as keyword and collocation analysis. It is now the official Python API to CWB.

[10] https://github.com/ausgerechnet/spheroscope.

match whether it is a true positive (if the tweet is contained in any gold standard), cf. Figure 2.



whole	0	1	tweet	TP ▲
NewStatesman : " None of the Brexit backing politicians would stop traffic because most people would like to run over them "	most people	to run over them	737596416470581249	True
&#x27; I want less red tape . But I would like the UK to stay in the EU &#x27; . ET news ed on #Brexit https://t.co/gSCEz6zx6y https://t.co/AuhIfiSSZr	I	the UK to stay in the EU	728891461853433857	True
I &#x27;d just like to say that I support the UK leaving the EU . #brexit #brexitorbust	I	to say that I support the UK leaving the EU	745247676871086081	True
As an Aussie in the UK , I &#x27;d like to have an opinion on #Brexit , but it &#x27;d be just like us in Eurovision : you heard it , but not sure why .	I	to have an opinion on #Brexit	730915874220167168	?

Fig. 2. Concordance View. For each query result, the actual text of the whole tweet is displayed, as well as the surface realizations of each defined slot. Additionally, column TP indicates whether the match is a *true positive* (True), *false positive* (False), or unknown (?) as the tweet is not in the gold standard.

4.2 Evaluation and Discussion

Results for our current version of the queries for *desire* and *membership* can be found in Table 2. Note that the most reliable estimates for precision can be taken from the annotation of actual query `matches`, whereas recall is most accurately estimated from `post` (since `pre` was used in the course of developing queries).

Table 2. Pattern-based evaluation of query approach for patterns *desire* and *membership* on different data sets alongside prevalence values. Recall of querying can most reliably be estimated from `post`, while precision can most reliably estimated on actual query matches (indicated in bold).

pattern	data set	prevalence	TN	FN	TP	FP	precision	recall	support
desire	`pre`	0.07	721	31	30	3	0.91	0.49	785
	`post`	0.05	923	25	19	6	0.76	**0.43**	973
	`matches`				2361	97	**0.96**		175022
membership	`pre`	0.10	705	62	13	5	0.72	0.17	785
	`post`	0.07	901	65	6	1	0.86	**0.08**	973
	`matches`				952	53	**0.95**		54412

As noted above, corpus queries are abstractions of the manually identified hits for a given pattern in the gold standard (based on `pre`). While they help us to find several hundred thousands of instances of *desire* on the corpus, their recall is restricted to maximize precision. In this section, we explore the nature of potential recall issues in more detail.

Statistical measures on precision and recall only show part of the picture: since our logical patterns are much more abstract than their realizations in the corpus, it is likely that, for politically relevant statements, even if an individual instance was missed, we may still have found other tweets containing equivalent information on the same entities and concepts. We therefore conducted a qualitative evaluation of tweets from our gold standard that were marked as *desire*, but were not retrieved by any of the queries written for this pattern.

We found a total of 65 false negatives for *desire* in our gold standard. Slightly more than half of these instances were excluded from further analysis because they were either no longer part of the corpus (9), they were assigned to subpatterns of *desire* (6), or because they were categorized as purely situational, e.g. because the entity was the speaker (22). The remaining 28 tweets were left as genuinely relevant statements to examine in more detail.

Typical reasons why a tweet was missed by the queries include both syntactic and lexical properties. On the syntactic side, we found long distances between the entity and the formula (**Banks** now call for 'passporting'; to be ditched and instead want a **'hard Brexit'**). Similar issues relate to tweets containing uncommon vocabulary or typos in slots using word lists (**Britian** want Brexit to go away).

In order to see whether tweets with equivalent content were present in the query results, we searched for the ENTITY for each of these false negatives within the query matches for *desire* and read the matches. For 23 out of 28 cases, an exact or very close match was found. For instance, while *Dennis Skinner for Brexit !!! YASSSSSSS !!!!* was missed, our *desire* queries matched *Dennis Skinner backs Brexit for democracy* and *Labour MPs Dennis Skinner and John Mann back Brexit.* Occasionally, the ENTITY was expressed in slightly different ways, but could still be related back to the same referent with contextual understanding. This incudes the following tweet, which we missed due to its relative clause: *Andrew Neil is chair of @spectator which has come out for #Brexit.* While our query results for *desire* did not include the @spectator account as the ENTITY, several instances referred to *The Spectator.*

For five relevant false negatives, we could not find a very similar equivalent in the query results. In some cases, the ENTITY was a relevant actor, but still infrequent in the corpus (*Globalists R desperate to abolish nations & families*). Alternatively, the FORMULA was too vague to be reconstructed (*You can sense people revelling in it on some level. Desperate for* ***something to come out that proves Farage or Leave or Brexit did this***).

In summary, this evaluation suggests that, at least for statements that have been expressed by a reasonably large number of users, the queries mostly still find logically equivalent propositions even where individual realisations are missed due to unusual wording or syntax.

5 Hierarchical Queries

5.1 Methods

Besides running on the overall corpus, queries can be nested to find argument fragments *within* the slots of larger fragments. For instance, if we run queries for the *membership* pattern on the FORMULA slot of our *desire* queries, we expect the results to be a hierarchical combination of the two patterns (e.g. *I want Britain to be in the EU*). The technical implementation consists of four steps:

1. Run queries for a given pattern and extract the text spans matching the slots of individual placeholders.
2. Form one sub-corpus per placeholder slot containing only these text spans.
3. Run queries for patterns of the correct sort on each sub-corpus.
4. Further instantiate the extracted formulae by substituting placeholders in the partial formula of the outer query with the formula obtained from the sub-query.

It is obvious that, in order for the results to be meaningfully interpretable, the inner query needs to be contained within the relevant slot of the outer query. However, it is less clear whether one should only consider exact matches (where the *membership* statement matches the entire FORMULA slot of *desire*) or also accept cases where only a part of the outer slot is matched by the inner query.

5.2 Evaluation

Therefore, we evaluated hierarchical queries for the combined pattern *ENTITY desires MEMBERSHIP*.

Table 3. Evaluation of hierarchical queries for *ENTITY desires MEMBERSHIP* for different positions of the inner query in the outer slot

inner/outer	#matches	TP	FP	correct example
exact	446	48	2	Donald Trump Supports ***The UK Leaving The European Union***
left	260	33	17	he would support ***Texas leaving the US*** *and becoming an independent state*
right	215	11	39	let's hope *we get a strong turnout on the day and* ***we leave the EU***
within	163	0	50	—

Table 3 shows the number of matches for each position of the inner query in the slot of the outer query, as well as the number of true and false positives in a manual validation of a random sample of 50 tweets for each position. *Exact* matches are almost always correct instances of the combined pattern. The majority of cases where the inner query match is only aligned with the *left* slot boundary are also correct, although the precision is considerably lower than for complete matches.

5.3 Discussion

A common type of false positive in this set of tweets is due to the ambiguity of the word *join*, e.g. *We wish the Netherlands will join us soon with a #Nexit and kick out their anti-democratic rulers.* Our combined query misinterprets this tweet as the Netherlands becoming a member of *us*. Most matches where the end of the inner query result is aligned with the *right* slot boundary are mostly false positives. Even though such cases usually do involve statements about membership, the membership assertion is typically embedded in some other statement that would also need to be parsed for a meaningful interpretation (e.g. *I really hope the Brits understand how turbulent Europe will be if **UK leaves EU***). Finally, none of the cases where the inner query match is strictly contained *within* the outer query's slot were true positives. Similarly to the *right* overlap, while the formula was typically related to membership, a multi-step parsing process involving additional patterns would have been required (*I wished I knew if **UK leaving the EU** is good or bad*).

6 Fine-Tuning LLMs

6.1 Methods and Evaluation

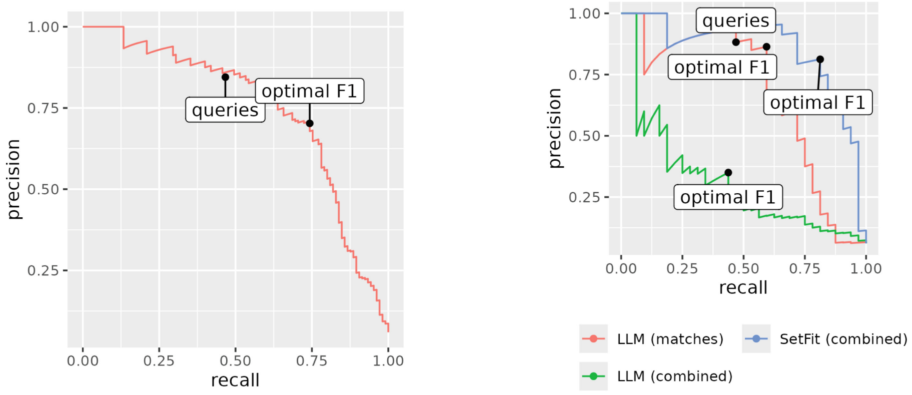
Supervised Prediction. Due to their low prevalence in the corpus, training an automatic system to detect individual argumentative fragments is challenging. The straightforward state-of-the-art approach is to fine-tune a large language model (LLM) on the manually annotated gold standard. The `combined` data set, comprising `pre` and `post`, cf. Section 3.2, consists of 1758 annotated tweets with 105 positive examples of *desire* (i.e. a prevalence of ca. 6%). Using 70% as training data and 30% as test data leaves us with 73 positive training examples (out of 1231 training examples), and 32 positive test examples (out of 527 test examples).[11]

In a first attempt, we use distilbert-base-uncased [29] as a base model and fine-tune using the `transformers` package with standard settings.[12] The trained model yields scores for both classes (*desire* and *no desire*); we focus on the positive class here. Note that scores for the two classes have a near-perfect negative correlation. A cut-off value for this score determines the trade-off between precision and recall; see Fig. 3 for the resulting precision-recall curves. A standard composite measure is the area under this curve (PR-AUC).[13] As can be seen from Fig. 3b (line *LLM (combined)*), the trained model performs poorly: precision values of, say, 50% can only be achieved with less than 25% recall (and vice

[11] All train/test splits are stratified random samples, i.e. we take random samples but make sure that the ratio of positive examples remains the same across splits.

[12] AutoModelForSequenceClassification, learning rate 2e-5, 5 epochs, 0.01 weight decay.

[13] Alternatively, we could look at the area under the receiver-operating characteristic (ROC) curve, which plots the true positive rate (precision) against the false positive rate (1 − specificity). This curve is however more suitable for situations where both classes (positive and negative) are equally prevalent or at least equally important.



(a) PR curve of LLM (trained on matches) on all of combined

(b) PR curve of different LLMs on test-split of combined.

Fig. 3. PR curves of LLMs on `combined` and its test split.

versa). The queries, on the other hand, achieve the expected high precision of 88% on the test set, and a stable trade-off with 47% recall.

As mentioned in the introduction, recent advancements in NLP have brought forth LLM frameworks that can generalise from very small numbers of training examples. Here, we use SetFit as such a few-shot classifier, and fine-tune the paraphrase-multilingual-mpnet-base-v2 model on our training examples. The PR curve of this approach outperforms the first attempt by a large margin (see Fig. 3b, line *SetFit (combined)*) and achieves competitive results compared to our query-based approach.

Generalizing from Query Matches. Additionally, we present an approach that leverages our corpus queries as training data for fine-tuning the LLM. We use all query matches, except for those in the `combined` gold standard to ensure comparability. We take 70% of a total of 145,699 matches for the *desire* pattern as positive training examples and add the same amount of random tweets (excluding query matches and those in `combined`) as negative examples. Note that for training, we assume all query matches to be instances of *desire* and randomly selected tweets to be negative examples. This is a reasonable approximation due to the high precision of the queries (ca. 96%) and the low prevalence of *desire* (ca. 6%).

Our approach can likely be improved considerably by optimizing any of the following parameters: Firstly, we could train on all query results. However, with the setting at hand, we can also evaluate how well the LLM predicts query results (see below for results). Secondly, we could provide a dataset with the (estimated) prevalence of *desire*. Lastly, we could try different base models and parameter settings (learning rate, weight decay, etc.). However, our goal here is a proof of concept, not engineering the best possible system.

Unsurprisingly, an LLM trained on query matches can accurately distinguish query matches from other tweets, i.e. it can learn the formal linguistic structures expressed by the queries. Recall that we only used 70% of the matches as positive training examples. Evaluating the LLM classifier on the remaining examples (mixed with random tweets) yields a PR-AUC of 0.9978. However, we are interested in its performance to detect the *desire* pattern, not just desire that is also captured by the queries (whose estimated recall is ca. 43%).

The precision-recall curve of this LLM on `combined` in Fig. 3a is thus more interesting. We also indicate the performance of the corpus queries in the graph. It is no coincidence that this data point lies on the PR curve of the LLM, which retrievs query matches nearly perfectly. At this point of the curve, the query matches and LLM predictions are almost identical. By lowering the LLM decision threshold, we move down the PR curve, improving recall at the cost of precision. Alternatively, we can further improve precision if we accept an even lower recall. Many reasonable trade-off points between precision and recall are available. In the graph, we also the trade-off that maximises F_1, i.e. the harmonic mean between precision and recall. We determine this value *ex post* for reasons of simplicity; in practical applications, it can also be determined on a separate development set.

Table 4. Comparison of different approaches to detect *desire* on the complete data set `combined` (top) and on test-split of `combined` (bottom). The query approach yields highest precision, the LLM trained on query matches can yield higher recall with with still decent precision values (as exemplified by the point of optimal F_1, indicated in bold).

data set	prev	approach	FN	FP	TN	TP	precision	recall	F_1
`combined`	0.06	LLM (matches)	28	33	1620	77	0.70	**0.73**	**0.72**
		queries	56	9	1644	49	**0.84**	0.47	0.60
test-split	0.06	LLM (matches)	9	6	489	23	0.79	0.72	0.75
		LLM (combined)	19	26	469	13	0.33	0.41	0.37
		SetFit (combined)	7	6	489	25	0.81	**0.78**	**0.79**
		queries	17	2	493	15	**0.88**	0.47	0.61

Figure 3b shows PR curves on the test set of `combined`, where we can compare the LLM trained on the train-split of `combined` (*LLM (random2000)*) with the one trained on query matches (*LLM (matches)*). The LLM trained on query matches is far superior to the one trained only on a couple of dozen of positive examples. Table 4 lists detailed results for all approaches on `combined` and its test-split (for LLMs, the decision threshold is set at the point of optimal F_1). In terms of precision, the corpus queries yield the best results (as by design).

However, the LLM trained on query matches can yield better recall, as is exemplified at the point of optimal F_1 of the PR curve.

6.2 Discussion: Qualitative Comparison of Approaches

As seen in Table 4, at the point of optimal F_1, the LLM approach trained on query matches achieves higher recall than the queries, but lower precision. In this section, we examine the differences between these two approaches through a qualitative analysis of tweets in the gold standards that were found by the LLM, but not by the queries.

The first group of tweets are newly identified true positives, i.e. tweets that contribute to the LLM achieving higher recall than the queries. The results suggest that the improvement in recall can be attributed to systematic factors. The main patterns in the true positives unique to the LLM include tweets containing typos (*Britian*) or short modifier phrases (*Denmark* ***for one*** *will be queuing up to leave*). While it would technically be possible to write queries that handle such cases, such modifications would either introduce unwarranted complexity for relatively small improvements, or they would reduce the queries' precision by introducing more opportunities for false positives.

Additionally, the queries rely on linguistic pre-processing, in particular on POS tagging. While this information is helpful in specifying grammatical patterns, tagging errors occasionally prevented the queries from finding relevant tweets. Thus, the LLM found several nominalizations that the POS tagger misinterpreted as adjectives, causing the query to miss a noun phrase (e.g. *The British want EU migrants to stay*). Similarly, the queries impose semantic restrictions via word lists where necessary, which obviously limits the scope of words that can possibly be matched in a given position. In contrast, the LLM found tweets with unusual entities like *noted Europhile paper backs Brexit.*

Finally, some hits found by the LLM contained syntactic patterns for which we had no queries – either because the expression contained non-standard syntax (*If we Brexit., ending the Barnet agreement, I'm for!*), or because the constructions were too rare to reasonably justify developing a manual query (*Very much looking forward to seeing nigel farage in action tonight*).

False positives (FP) unique to the LLM were usually syntactically similar to one of the queries, but did not match the correct semantics (*#Brexit gloom is for losers*). In rarer cases, the tweets contained some reference to desire that was too implicit according to the guidelines (*"Being pro brexit is wacist!" said the hipster white brits to the black brits* – this tweet is not considered *desire* since it is a general statement rather than a specific entity desiring something).

7 Limitations

The case study currently pursues a comparatively narrow topical focus; the generalizability of our findings remains to be explored. Scaling the overall approach to large repositories of logical patterns is possible in principle but resource-intensive: Firstly, the method relies on the manual development of corpus queries, which involves corpus-linguistic analysis, and secondly, query development needs manual annotation of random samples to find suitable starting points (however, queries then generalise from very few examples). The task of annotating samples for matches with logical patterns is conceptually difficult, and agreement between human annotators is comparatively low (with notably higher agreement on the `post` dataset). Our approach to fine-tuning LLMs using query results is currently at the proof-of-concept stage and could likely be substantially improved in further work.

8 Conclusion

We have described an approach to extracting argument fragments from short text snippets on social media, using corpus queries to fill slots in predefined logical patterns. Patterns and queries can be applied in a nested fashion, allowing for the extraction of more complex semantic content. We have demonstrated an application of our methodology in the generation of training data for use in the fine-tuning of LLMs. Without any manual annotation, we achieve comparable results to state-of-the-art few-shot learning approaches such as SetFit that have been trained on more than 1200 manually annotated tweets.

Ongoing efforts aim to conduct automated logical reasoning steps over the extracted argument fragments, which will require use of semantic similarity measures. Moreover, we intend to extend the scope of the method both w.r.t. supported lanuages and w.r.t. the length and degree of coherence of the underlying text, covering also, e.g., newspaper articles or parliamentary debates, and aiming to extract argumentation chains instead of just argument fragments.

Acknowledgements. The work has been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 377333057 "Reconstructing Arguments from Newsworthy Debates (RAND)".

References

1. Alsinet, T., Argelich, J., Béjar, R., Cemeli, J.: A distributed argumentation algorithm for mining consistent opinions in weighted Twitter discussions. Soft. Comput. **23**(7), 2147–2166 (2019). https://doi.org/10.1007/s00500-018-3380-x
2. Beck, T., Lee, J.U., Viehmann, C., Maurer, M., Quiring, O. and Gurevych, I.: Investigating label suggestions for opinion mining in german covid-19 social media (2021)
3. Bhatti, M.M.A., Ahmad, A.S., Park, J.: Argument Mining on Twitter: a case study on the planned parenthood debate. In: Proceedings of the 8th Workshop on Argument Mining, pp. 1–11. Association for Computational Linguistics, Punta Cana, Dominican Republic (2021) https://doi.org/10.18653/v1/2021.argmining-1.1 https://doi.org/10.18653/v1/2021.argmining-1.1

4. Bosc, T., Cabrio, E., Villata, S.: Tweeties squabbling: positive and negative results in applying argument mining on social media. In: Computational Models of Argument, COMMA 2016. Frontiers Artificial Intelligence Applications, pp. 21-32. IOS Press (2016)
5. Cabrio, E., Villata, S.: Five years of argument mining: a Data–driven Analysis. In: International Joint Conference on Artificial Intelligence, IJCAI 2018, pp. 5427-5433. ijcai.org (2018)
6. Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V.: GATE: an Architecture for Development of Robust HLT applications. In: Annual Meeting of the Association for Computational Linguistics, ACL 2002, pp. 168-175 (2002). https://doi.org/10.3115/1073083.1073112
7. Dusmanu, M., Cabrio, E., Villata, S.: Argument mining on Twitter: arguments, facts and sources. In: Empirical Methods in Natural Language Processing, EMNLP 2017, pp. 2317-2322. ACL (2017)
8. Dykes, N., Evert, S., Göttlinger, M., Heinrich, P., Schröder, L.: Argument parsing via corpus queries. Inf. Technol. **63**(1), 31–44 (2021). https://doi.org/10.1515/itit-2020-0051
9. Dykes, N., Evert, S., Göttlinger, M., Heinrich, P., Schröder, L.: Reconstructing arguments from noisy text: introduction to the RANT project. Datenbank- Spektrum **20**, 123–129 (2020)
10. Evert, S., Hardie, A.: Twenty-first century Corpus Workbench: updating a query architecture for the new millennium. In: Corpus Linguistics, CL 2011. University of Birmingham (2011)
11. Evert, S.: The CWB development team: the IMS Open Corpus Workbench (CWB) CQP Interface and Query Language Tutorial. CWB Version 3.5. 2022. https://cwb.sourceforge.io/documentation.php
12. Feger, M., Dietze, S.: TACO–Twitter Arguments from COnversations. (2024)
13. Feng, S.Y., et al.: A Survey of data augmentation approaches for NLP. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 968-988. Association for Computational Linguistics, Online (2021). https://doi.org/10.18653/v1/2021.findings-acl.84
14. Goudas, T., Louizos, C., Petasis, G., Karkaletsis, V.: Argument extraction from News, blogs, and Social Media. In: Artificial Intelligence: Methods and Applications, SETN 2014, pp. 287-299. Springer (2014)
15. Grosse, K., Chesñevar, C., Maguitman, A., Estevez, E.: Empowering an eGovernment platform through Twitter-based arguments. Inteligencia Artif. **15**(50), 46–56 (2012)
16. Hardie, A.: CQPweb - combining power, flexibility and usability in a corpus analysis tool. Int. J. Corpus Ling. **17**(3), 380–409 (2012)
17. Humml, M., Schröder, L.: Common Knowledge of abstract groups. In: AAAI Conference on Artificial Intelligence (AAAI 2023), pp. 6434-6441 (2023). https://doi.org/10.1609/aaai.v37i5.25791
18. Liu, H., et al.: Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning. arXiv preprint arXiv:2205.05638 (2022). https://doi.org/10.48550/arXiv.2205.05638
19. Lytos, A., Lagkas, T., Sarigiannidis, P., Bontcheva, K.: The evolution of argumentation mining: from models to social media and emerging tools. Inf. Process. Manage. **56**(6), 102055 (2019). https://doi.org/10.1016/j.ipm.2019.102055
20. Lytos, A., Lagkas, T., Sarigiannidis, P.G., Argyriou, V., Eleftherakis, G.: Modelling argumentation in short text: a case of social media debate. Simul. Model. Pract. Theory **115**, 102446 (2022). https://doi.org/10.1016/J.SIMPAT.2021.102446

21. Minnen, G., Carroll, J., Pearce, D.: Applied morphological processing of English. Nat. Lang. Eng. **7**(3), 207–223 (2001)
22. Owoputi, O., O'Connor, B., Dyer, C., Gimpel, K., Schneider, N., Smith, N.: Improved part-of-speech tagging for online conversational text with word clusters. In: Human Language Technologies, HLT-NAACL 2013, pp. 380-390. ACL (2013)
23. Pantel, P., Pennacchiotti, M.: Espresso: leveraging generic patterns for automatically harvesting semantic relations. In: Computational Linguistics / Annual Meeting of the Association for Computational Linguistics, ACL 2006. ACL (2006)
24. Proisl, T., Uhrig, P.: SoMaJo: state-of-the-art tokenization for German web and social media texts. In: Proceedings of the 10th Web as Corpus Workshop (WAC-X) and the EmpiriST Shared Task, pp. 57-62. Association for Computational Linguistics, Berlin (2016). https://doi.org/10.18653/v1/W16-2607
25. Qiu, Y., Jin, Y.: ChatGPT and finetuned BERT: a comparative study for developing intelligent design support systems. Intell. Syst. Appl. **21**, 200308 (2024). https://doi.org/10.1016/j.iswa.2023.200308
26. Rahman, A.M.M., Yin, W., Wang, G.: Data augmentation for text classification with EASE. In: Abbas, M., Freihat, A.A. (eds.) Proceedings of the 6th International Conference on Natural Language and Speech Processing (ICNLSP 2023), pp. 324-332. Association for Computational Linguistics, Online (2023)
27. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using siamese BERT-Networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics (2019)
28. Ritter, A., Mausam, Etzioni, O., Clark, S.: Open domain event extraction from twitter. In: Knowledge Discovery and Data Mining, KDD 2012, pp. 1104- 1112. ACM (2012)
29. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. CoRR abs/1910.01108 (2019)
30. Schaefer, R., Stede, M.: Argument mining on Twitter: a survey. Inf. Technol. **63**(1), 45–58 (2021). https://doi.org/10.1515/itit-2020-0053
31. Shnarch, E.,et al.: Will it Blend? blending weak and strong labeled data in a neural network for argumentation mining. In: Gurevych, I., Miyao, Y. (eds.) Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 599-605. Association for Computational Linguistics, Melbourne, Australia (2018). https://doi.org/10.18653/v1/P18-2095
32. Son, Y., et al.: Recognizing counterfactual thinking in social media texts. In: Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers) (2017). https://doi.org/10.18653/v1/p17-2103
33. Tunstall, L., et al.: Efficient few-shot learning without prompts. arXiv preprint arXiv:2209.11055 (2022). https://doi.org/10.48550/ARXIV.2209.11055
34. Walton, D., Reed, C., Macagno, F.: Argumentation Schemes. Cambridge University Press (2008). https://doi.org/10.1017/CBO9780511802034

Computational Models of Argumentation

Enhancing Abstract Argumentation Solvers with Machine Learning-Guided Heuristics: A Feasibility Study

Sandra Hoffmann(✉), Isabelle Kuhlmann, and Matthias Thimm

FernUniversität in Hagen, Hagen, Germany
{sandra.hoffmann,isabelle.kuhlmann,matthias.thimm}@fernuni-hagen.de

Abstract. Abstract argumentation frameworks model arguments and their relationships as directed graphs, often with the goal of identifying sets of arguments capable of defending themselves against external attacks. The determination of such admissible sets, depending on specific semantics, is known to be an NP-hard problem. Recent research has demonstrated the efficacy of machine learning methods in approximating solutions compared to exact methods. In this study, we leverage machine learning to enhance the performance of an exact solver for credulous reasoning under admissibility in abstract argumentation.

More precisely, we first apply a random forest to predict acceptability, and subsequently use those predictions to form a heuristic that guides a search-based solver. Additionally, we propose a strategy for handling varying prediction qualities. Our approach significantly reduces both the number of backtracking steps and the overall runtime, compared to standard existing heuristics for search-based solvers, while still providing a correct solution.

Keywords: Abstract argumentation · Heuristics · Random forest

1 Introduction

Argumentation is central for human communication and interaction, hence there are various strategies of implementing this concept in approaches to artificial intelligence. In the field of *abstract argumentation* [8], the underlying idea is to focus on the interplay between arguments and counterarguments rather than on the content of the arguments themselves.

The core formalism in this field is the *abstract argumentation framework*, which can be understood as a directed graph in which the nodes represent the given arguments, and the edges represent an attack relation between them.

Figure 1 shows an example of such a framework. Semantics are commonly expressed through so-called *extensions*, which are sets of arguments that jointly fulfill certain conditions. A fundamental semantics in the field of abstract argumentation is the concept of admissibility. In order to be an admissible extension,

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 185–201, 2024.
https://doi.org/10.1007/978-3-031-63536-6_11

arguments in the set must not attack each other (i. e., the set must be *conflict-free*) and they have to defend each other from all outside attacks.

Typical problems in abstract argumentation include the problem of deciding whether an argument is included in at least one extension (or all extensions) wrt. a specific semantics, or the problem of determining an extension or enumerating all of them wrt. a specific semantics.

The literature already provides different families of (exact) approaches to solve the above-mentioned reasoning problems in abstract argumentation. One such family consists of reduction-based approaches—see, e. g., [1,9,17,21,24]—which encode a given problem in a different formalism—e. g., as a Boolean satisfiability problem—and then use an existing solver for that formalism. Another family of approaches consists of backtracking-based methods that make use of heuristics to guide the search procedure—see, e. g., [12,22,23].

Since most of the reasoning problems in abstract argumentation are computationally hard [10], this can result in exceedingly long runtimes when using an exact algorithm. To counteract this issue, machine learning-based approaches have been proposed in the literature [6,7,15,19]. However, although these approaches proved to be significantly faster than their exact counterparts, they are not guaranteed to yield correct results (for a deeper analysis, see also [16]). Thus, the main advantage of an exact method (such as a reduction- or backtracking-based approach) is that it always provides correct results, while the main advantage of a (purely heuristic) machine learning-based approach is its runtime performance. An approach for combining these advantages is the use of machine learning techniques to predict the "best" exact solver from a portfolio [14,25]. In the work at hand, we aim to harness the advantages of machine learning methods in a different manner. More precisely, we use predictions made by a machine learning model in order to inform a heuristic that guides a backtracking-based approach which ultimately yields a correct result. As an example for the overall approach we consider the task of deciding whether a given argument is accepted under admissibility [8], which is a core aspect in many reasoning problems.

In an experimental evaluation we compare the use of our machine learning-based heuristic (using a random forest) to the standard heuristic of the backtracking-based solver Heureka [12], and we demonstrate that both the number of backtracking steps as well as the overall runtime can be reduced when our newly proposed heuristic is applied.

To summarize, our contributions are as follows:

- We present an approach that exploits the strengths of both machine learning and reasoning techniques by using machine learning-based predictions to create a heuristic which can accelerate an exact, backtracking-based solver.
- Our approach offers a flexible solution, as both the machine learning component and the backtracking-based solver can be specified as desired.
- In an experimental analysis, we show that our approach leads to a significant decrease in both the number of backtracking steps and overall runtime, when compared to a standard heuristic.

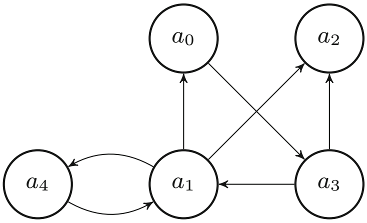


Fig. 1. An abstract argumentation framework.

The remainder of the paper is organized as follows. In Sect. 2 we provide some preliminaries on the topic of abstract argumentation. After giving an overview on current solution approaches for problems in abstract argumentation in Sect. 3, we propose a machine learning-guided heuristic in Sect. 4. An extensive experimental analysis is presented in Sect. 5, Sect. 6 details the limitations of our research and finally we conclude in Sect. 7.

2 Preliminaries

An *abstract argumentation framework* (AF) [8] is a tuple $F = (\mathsf{Args}, R)$, with Args being a set of arguments and $R \subseteq \mathsf{Args} \times \mathsf{Args}$ defining an attack relation. An argument $a \in \mathsf{Args}$ *attacks* another argument $b \in \mathsf{Args}$ if $(a, b) \in R$. On the other hand, an argument $a \in \mathsf{Args}$ is *defended by* a set of arguments $E \subseteq \mathsf{Args}$ if for all $b \in \mathsf{Args}$ with $(b, a) \in R$, there exists a $c \in E$ with $(c, b) \in R$.

An *extension* is a set of arguments that are jointly acceptable, given a set of conditions. Exactly which conditions need to be satisfied is determined by a *semantics*. There exists a multitude of different semantics in the literature, however, we focus on the *preferred* semantics introduced in the seminal paper by Dung [8].

Definition 1. *Let $F = (\mathsf{Args}, R)$ be an argumentation framework. A set $E \subseteq \mathsf{Args}$ is*

- *conflict-free if there are no $a, b \in E$ such that $(a, b) \in R$,*
- *admissible if E is conflict-free and each $a \in E$ is defended by E within F,*
- *complete if every argument $a \in \mathsf{Args}$ defended by E is also included in E,*
- *preferred if E is a $\subseteq$-maximal complete extension, and*
- *grounded if E is a $\subseteq$-minimal complete extension.*

Note that the grounded extension is uniquely determined [8].

Typical decision problems in the area of abstract argumentation include the problem of deciding whether a given argument is included in at least one extension (*credulous acceptability*) or all extensions (*skeptical acceptability*) wrt. a given semantics. In the following, we denote the problem of deciding credulous acceptability wrt. preferred semantics as DC. Note that this problem is equivalent to the problems of deciding credulous acceptability under admissible, and under complete semantics.

Algorithm 1: Backtracking-based algorithm SEARCH for checking credulous acceptance wrt. admissibility

```
Data: $F = (\mathsf{Args}, R)$, $S_{in}, S_{out} \subseteq \mathsf{Args}$
Result: TRUE if there is admissible $S'$ with $S_{in} \subseteq S'$.
if $S_{in}$ is not conflict-free then
    return FALSE
if $S_{in}$ is admissible then
    return TRUE
Pick $a \in \mathsf{Args} \setminus (S_{in} \cup S_{out})$
return SEARCH$(F, S_{in} \cup \{a\}, S_{out})$ OR SEARCH$(F, S_{in}, S_{out} \cup \{a\})$
```

Example 1. Consider the AF illustrated in Fig. 1. The conflict-free sets of this AF are

$$\{\emptyset, \{a_0\}, \{a_1\}, \{a_2\}, \{a_3\}, \{a_4\},$$
$$\{a_0, a_2\}, \{a_0, a_4\}, \{a_2, a_4\}, \{a_3, a_4\},$$
$$\{a_0, a_2, a_4\}\}.$$

Out of these sets, only $\emptyset$, $\{a_4\}$, $\{a_0, a_4\}$, and $\{a_0, a_2, a_4\}$ defend themselves, and are thus admissible. Further, the only complete sets are $\emptyset$ and $\{a_0, a_2, a_4\}$, which makes the grounded extension (i.e., the $\subseteq$-minimal complete extension) $\emptyset$, and the set of preferred extensions (i.e., the $\subseteq$-maximal complete extensions) consists only of $\{a_0, a_2, a_4\}$. We can also see that the set of arguments contained in at least one admissible set (i.e., the set of credulously acceptable arguments wrt. admissibility) is $\{a_0, a_2, a_4\}$, which is equal to the set of credulously accepted arguments under complete or preferred semantics.

3 Solution Approaches in Abstract Argumentation

The methods employed to address decision problems in abstract argumentation can be broadly categorized into reduction-based or direct approaches [5]. Reduction-based solvers operate by translating the reasoning problem into other formalisms, such as answer-set programming [9,11], constraint-satisfaction problems [2,4] or Boolean satisfiability [21,26], leveraging existing solvers in those domains. The advantage of the reduction-based approach lies in the high efficiency of these existing solvers. On the other hand, direct approaches involve the implementation of a dedicated algorithm tailored to the structure of AFs, often utilizing backtracking. Direct solvers retain the structural information of the AF, allowing them to exploit specific shortcuts relevant to certain semantics [5].

Algorithm 1 describes a simple backtracking strategy to assess argument justification. Given an AF F and two argument sets S_{in} and S_{out}, the algorithm recursively explores potential admissible sets by considering the inclusion or exclusion of individual arguments. It terminates and returns False if S_{in} is not conflict-free, ensuring the absence of internal attacks. If S_{in} is already admissible, the algorithm returns True. Otherwise, it selects an argument a from the remaining set of arguments and recursively follows two branches: one including a in S_{in} and maintaining S_{out}, and another excluding a from S_{in} and incorporating a into S_{out}. If the search in the first branch succeeds the second branch does not have to be explored. If the search in the first branch fails, we say that the algorithm *backtracks* and it is required to continue with the second branch. The algorithm returns True if either branch results in an admissible set. The algorithm can determine if an argument a is contained in at least one admissible/preferred/complete extension by calling it with $\mathsf{SEARCH}(F, \{a\}, \emptyset)$.

The order in which arguments are processed—i. e. how argument a is determined in line 5 of Algorithm 1—plays a crucial role in the algorithm's performance, and different heuristics can be employed for this purpose. The algorithm we use in our study specifies the order in a deterministic way. The selected heuristic calculates a confidence value for each argument. Subsequently, these values are arranged in descending order to establish the total ordering.

Example 2. Consider again the AF in Fig. 1, which depicts an AF with the preferred extension $\{a_0, a_2, a_4\}$, and the task to decide DC wrt. a_2. Assuming the order determined by a certain heuristic is $(a_1, a_3, a_4, a_0, a_2)$[1], the binary tree visualizing the recursive calls needed to solve this task using Algorithm 1 has a depth of 4. In contrast, building the order based on a perfect prediction of each argument's acceptability yields a depth of 2, thereby enhancing the algorithm's efficiency.

Note that Algorithm 1 only showcases the general principle of search-based algorithms. Existing search-based solvers [12,22,23] are more involved and rely on similar techniques as DPLL- and CDCL-solvers from satisfiability solving [3].

4 Machine Learning-Guided Heuristics

The goal of this paper is to improve the performance of a direct solver by reducing the number of backtracking steps necessary to decide DC wrt. a given argument.

In order to do so, we employ a machine learning classifier and use the obtained predictions to guide a direct solver. For our research, we decided to predict the overall acceptance status of an argument and use this prediction, along with a confidence measure, to build our heuristic. This heuristic determines the order in which the search algorithm processes the arguments. One might question why we did not employ the classifier to directly predict the optimal order for each argument, thereby eliminating the need for a priority heuristic altogether. However, it is crucial to acknowledge that for each argument, there exist multiple ideal orderings that would effectively guide the solver.

Returning to Example 2, we determined a_0, a_2, a_4 to be the preferred extension of this AF. However, when deciding DC with respect to a_2, it does not matter whether we first pass a_0 or a_4 to the algorithm. Another possible approach

[1] This is the order determined by the standard heuristic used in Heureka [12].

would be to aim to directly predict admissible sets. The author in [18] describes a similar approach by training a graph neural network to predict which arguments are jointly admissible and then use this information to guide a SAT-based solver. While this approach yielded promising results and provides opportunities for further study, it also requires extensive neural network training, whereas our goal was to investigate whether a lightweight solution could already provide substantial improvements. More information on the training process is provided in Sect. 5.1.

We pass the obtained prediction outcomes as input to a direct solver and use them to develop a heuristic that prioritizes arguments based on their predicted acceptability. Arguments predicted with higher confidence to be accepted wrt. DC are processed first, while those likely to be rejected are processed last. We compare the results to those obtained using a heuristic that has demonstrated effective performance for DC in prior research [12].

Following an analysis of some initial experiments, we refine our approach by crafting a heuristic tailored specifically to the query argument. Subsequently, we assess the performance on further datasets.

To determine whether an argument a is acceptable wrt. DC, we need to find a preferred extension that contains a. In contrast, to prove that a is not acceptable wrt. DC, we need to establish that there cannot be a preferred extension containing a. As any conflict with the grounded extension signifies the rejection of a, our strategy in aiming to prove that a is not acceptable wrt. DC revolves around assuming a, as well as all arguments belonging to the grounded extension, are acceptable wrt. DC and devising a heuristic prioritizing arguments likely not being accepted. This approach aimed to ensure conflicts happen early on in the justification process, in order to enhance overall performance. While initial experiments showed promising outcomes in AFs with substantial grounded extensions, the efficacy diminished in AFs with an empty grounded extension. Even when restricting the heuristic's application only to AFs with non-empty grounded extensions, the results failed to significantly surpass those of the standard heuristic. This may be attributed to the fluctuation in prediction accuracy when considering related arguments. As detailed in Sect. 5, although our RF-model is able to classify most arguments correctly, when constructing a heuristic centered on a specific argument, substantial penalties can occur for inaccuracies in predicting related arguments. Additional research is needed to devise a robust heuristic for rejected arguments. Consequently, we exclusively consider accepted arguments in the subsequent sections of this paper.

5 Experimental Analysis

The following section offers an overview of the datasets we utilized, outlines our experimental setup, and presents the results obtained from our experiments.

5.1 Datasets and Setup

To conduct our experiments, we utilized the *kwt-train* and *kwt-test* datasets generated using the *KwtDungTheoryGenerator*[2] as described in [16].

Each graph in these datasets consists of 151 arguments, and the training and test set each contain 1000 graphs.

To assess the performance of our heuristic on larger datasets, we employed the *KwtDungTheoryGenerator* to generate a more extensive dataset called *kwt-large*. This new dataset comprises 10,000 graphs designated for training (*kwt-large-train*) and an additional 1000 graphs reserved for testing (*kwt-large-test*). The graphs within this dataset span a range of 100 to 300 arguments, with a total of 148,483 accepted arguments within the *kwt-large-test* set.

In our research, the primary emphasis lies on enhancing the performance of arguments that are credulously accepted. Accordingly, we sought to employ a graph type that featured a substantial number of accepted arguments for our third dataset. To achieve this, we harnessed the *AFBenchGen* graph generator[3] to create a supplementary set of 10,000 Barabasi graphs for training (*Barabasi-train*) and an additional 1,000 Barabasi graphs for testing (*Barabasi-test*). These graphs encompassed argument quantities ranging from 100 to 500, resulting in a total of 252,502 accepted arguments within the *Barabasi-test* set[4].

Previous research has suggested that standard machine learning classifiers are useful in predicting the acceptability status of arguments in an AF [13,16]. In [13] random forest (RF) classifiers trained using a comprehensive feature set provided the best results. This feature set comprised 10 node- and graph-based properties, namely the degree, closeness, Katz [20], and betweenness centrality as well as the number of the strongly connected components (SCC) of the AF, the size of the SCC each argument is part of, the average degree of the AF and whether it is irreflexive, strongly connected or aperiodic.

Building on these results, we trained individual RF classifiers for each dataset. The training and testing procedures were executed using Python, making use of the *scikit-learn*[5] and *networkx*[6] libraries. For a detailed overview of all three datasets, please refer to Table 1. To quantify the efficacy of our classification results, we use the standard metrics of *accuracy*, *recall* (also referred to as *true positive rate* (TPR)), *specificity* (also referred to as *true negative rate* (TNR)), and *precision*, as well as the *Matthews Correlation Coefficient* (MCC). We define a *true positive* (TP) as an argument in an AF that is accepted wrt. DC and was correctly classified as such. Accordingly, a *true negative* (TN) is a non-accepted argument that is correctly classified as such, and *false positives/negatives* (FP/FN) are the corresponding falsely classified counterparts.

[2] http://tweetyproject.org/r/?r=kwt_gen.

[3] https://sourceforge.net/projects/afbenchgen/.

[4] The datasets, the enhanced Heureka code and the individual results are available here: http://mthimm.de/misc/hkt_ratio24.zip.

[5] https://scikit-learn.org/stable/.

[6] https://networkx.org/.

Table 1. Overview of the *kwt*, *kwt-large* and *Barabasi* datasets.

Dataset	No of graphs	No of nodes	YES nodes	NO nodes
kwt-train	1,000	151,000	113,539	37,461
kwt-test	1,000	151,000	112,909	38,091
kwt-large-train	10,000	2,210,000	1,574,194	635,806
kwt-large-test	1,000	220,342	148,483	71,859
Barabasi-train	10,000	3,000,000	2,524,352	475,648
Barabasi-test	1,000	300,000	252,502	47,498

Table 2. Results for classifying the *kwt*, *kwt-large* and *Barabasi* test sets using an RF classifier trained on a total of 10 graph features.

Dataset	MCC	Accuracy	Recall (TPR)	Specificity (TNR)	Precision
kwt-test	0.987	0.995	0.994	1	1
kwt-large-test	0.990	0.996	0.994	1	1
Barabasi-test	0.792	0.947	0.980	0.771	0.958

Accuracy is defined as $\frac{\mathsf{TP}+\mathsf{TN}}{\mathsf{TP}+\mathsf{TN}+\mathsf{FP}+\mathsf{FN}}$, precision as $\frac{\mathsf{TP}}{\mathsf{TP}+\mathsf{FP}}$, TPR as $\frac{\mathsf{TP}}{\mathsf{TP}+\mathsf{FN}}$, TNR as $\frac{\mathsf{TN}}{\mathsf{TN}+\mathsf{FP}}$, and MCC as $\frac{\mathsf{TP}\cdot\mathsf{TN}-\mathsf{FP}\cdot\mathsf{FN}}{\sqrt{(\mathsf{TP}+\mathsf{FP})\cdot(\mathsf{TP}+\mathsf{FN})\cdot(\mathsf{TN}+\mathsf{FP})\cdot(\mathsf{TN}+\mathsf{FN})}}$.

We decided on using the Heureka solver [12] due to its implementation of a direct solution approach and its flexibility in incorporating custom heuristics to determine the order of arguments. The objective of the heuristic is to assign a real-number value to each argument through a mapping function. A higher value indicates a higher priority for a particular argument, influencing its processing order in the justification process. Specifically for DC, Heureka employs a standard heuristic that emphasizes arguments within strongly connected components, combining this with a path-based component.

In our experiments, Heureka is executed on each argument within our test sets, capturing both runtime and backtracking steps for individual arguments. The standard heuristic serves as a benchmark for comparing the outcomes of our experiments. To control the overall runtime for each dataset, a timeout of 10 minutes per argument is implemented.

5.2 Initial Experimental Analysis

We begin our experiments by training an RF classifier for each dataset. An overview of the classification metrics is provided in Table 2.

Our initial approach involves simply prioritizing the arguments predicted to be acceptable wrt. DC. The order of argument processing is determined by calculating a score for each argument based on the percentage of trees that favor the assigned label. If an argument is predicted to not be contained in any

Table 3. Results for classifying DC arguments in the kwt Dataset using the standard Heureka heuristic as well as a simple prediction-based ordering

MCC	Standard Backtracks	Prediction Backtracks	no of AFs
>0.7	39,919	1,082,091	8
>0.8	168,312	50,328,039	58
>0.9	10,033	1,531,500	33
1	280,676	0	898
Total	498,940	52,941,630	997

extension, we prioritize arguments with predictions that are close to the decision boundary.

We evaluate all arguments that are acceptable wrt. DC in the *kwt-test* set using both the prediction-based ordering and the standard heuristic.

The results, presented in Table 3, indicate that the simple ordering we employed successfully reduced the need for backtracking in cases with relatively accurate predictions, however, this approach severely penalizes wrong predictions, which led to an overall increase in backtracking steps. Additionally, using the simple ordering heuristic, Heureka was unable to solve three AFs within the 10-minute time limit per argument.

To gain deeper insights into the limitations of this approach, we conducted a detailed analysis of the AF that required the highest number of backtracking steps. While applying the prediction-based ordering resulted in a staggering 21,482,709 backtracking steps, the standard heuristic was able to resolve this AF without any backtracking.

Upon closer examination, we discovered that out of the 151 arguments in this AF, only 9 specific arguments were responsible for all the backtracking steps. These 9 arguments were the sole accepted arguments that were erroneously predicted as not accepted by our model. Furthermore, all of these arguments belonged to the same extension, and critically, none of them belonged to any other extension. As a result, these crucial arguments, which would be highly valuable for guiding our search algorithm, ended up being processed toward the end of the solving process. Consequently, a straightforward ordering approach proved to be insufficient. To reduce the overall number of required backtracking steps, a more refined heuristic is needed.

To establish whether an argument a is acceptable wrt. DC, we must identify a preferred extension E that contains a. Therefore, we want to prioritize arguments that are most likely part of E. We thus separate our AF into three distinct sets: Arguments that are likely not in E (*outExt*), arguments that defend a (*defenders*) and thus have the highest chance to be in E, and arguments that might be in E (*possibleExt*). Our refined algorithm starts by adding all arguments that are in conflict with a to the *outExt* set. Likewise, arguments predicted to be outside any extension are categorized within *outExt*. Subsequently, we then iterate through the remaining arguments, identifying whether arguments act as defenders of a by attacking its attackers or whether they undermine E by targeting arguments likely to be part of E. Arguments that do not fall into the categories of defenders or offenders are placed in the *possibleExt* set. Once all arguments are processed we determine the heuristic order, making sure, that all *defenders* are processed first. This is ensured by multiplying the prediction probability of each argument in a set by a dedicated factor for said set. Let the factors used to multiply the prediction confidence values be denoted as $\mathsf{x}, \mathsf{y}, \mathsf{z}$ for the *defenders*, *possibleExt* and *outExt* sets, respectively. The actual value of the factors is arbitrary, as long as the following conditions hold: $\mathsf{x} > \mathsf{y}$ and $\mathsf{z} > 1$. For more detailed information, please refer to Algorithm 2. In our experiments we set $\mathsf{x} = 1000$, $\mathsf{y} = 100$, and $\mathsf{z} = 2$.

Algorithm 2: MLPred Heuristic for accepted Arguments

```
Data: AF $aaf$, Prediction $pred$, Query Argument a
Result: Heuristic $h$
$attackRelation \leftarrow$ AttackRelation($aaf$);
$attackers \leftarrow attackRelation$.attacker_set($a$);
$attackeds \leftarrow attackRelation$.attacked_set($a$);
$outExt \leftarrow attackers \cup attackeds$;
$possibleExt \leftarrow itemIndex$;
$defenders \leftarrow \emptyset$;
for $i = 0; i <$ pred.args.size()-1; $i++$ do
    $curAttackeds \leftarrow attackRelation$.attacked_set($i$);
    $argIsDefender \leftarrow curAttackeds \cap attackers$;
    $argIsOffender \leftarrow curAttackeds \cap possibleExt$;
    if pred.predictLabel[$i$] == YES then
        if argIsOffender then
            $outExt \leftarrow i$;
            continue;
        else if argIsDefender then
            $defenders \leftarrow i$;
        else
            $possibleExt \leftarrow i$;
        $attackers \leftarrow attackRelation$.attacker_set($i$);
        $outExt \leftarrow attackers \cup curAttackeds$;
    else
        $outExt \leftarrow i$;
for arg in defenders do
    h.order[arg] $\leftarrow$ pred.predProb[arg] $* \mathsf{x}$
for arg in possibleExt do
    h.order[arg] $\leftarrow$ pred.predProb[arg] $* \mathsf{y}$
for arg in outExt do
    h.order[arg] $\leftarrow$ pred.predProb[arg] $* \mathsf{z}^{-1}$
```

Table 4. Results for classifying DC arguments in the kwt Dataset using the standard Heureka heuristic as well as the MLPred heuristic explained in Algorithm 2.

MCC	Standard Backtracks	Prediction Backtracks	no of AFs
>0.7	39,919	358,246	8
>0.8	185,587	4,320,665	60
>0.9	10,101	1,412,926	34
1	280,676	0	898
Total	516,283	6,091,837	1000

Table 5. Results for classifying the kwt Dataset using the standard Heureka heuristic as well as the MLPred heuristic explained in Algorithm 2 with a threshold of 0.35.

MCC	Standard Backtracks	Prediction Backtracks	no of AFs
>0.7	39,919	31,976	8
>0.8	185,587	226,790	60
>0.9	10,101	8,029	34
1	280,676	0	898
Total	516,283	266,795	1000

Running Heureka using this refined approach yielded a significant reduction in backtracking, nearly reaching a 90% reduction, and enabling Heureka to successfully solve all argumentation AFs within the allocated time, as shown in Table 4. However, when evaluated against the standard heuristic, it is evident that the total number of backtracking steps, though significantly improved, still falls short of matching the performance of the standard heuristic.

Within our dataset, all instances of backtracking occur in AFs where the predictive accuracy is not perfect. As we have observed during our in-depth analysis of an individual AF, the quality of predictions can vary not only between AFs but also among arguments within the same AF. Therefore, we require a method to make an informed choice of whether we can rely on the predictions generated by the machine learning model to effectively guide Heureka.

In our algorithm, the *defenders* set comprises the most critical arguments, as these directly support our query argument a. We operate on the assumption that a larger *defenders* set implies a more informative prediction for guiding Heureka. We also employ a threshold parameter below, which we opt to use the standard heuristic instead of the prediction. More specifically, this threshold dictates the required size of the *defenders* set in relation to the *possibleExt* set. In our experiments, we employed a threshold of 0.35. Re-running Heureka with this threshold produced the results presented in Table 5.

By implementing the threshold to filter out uninformative predictions, we successfully reduced the number of backtracking steps by nearly 50%. In the following section we will evaluate our initial results using larger, more diverse datasets.

Table 6. Results for classifying the *kwt-large* dataset using the standard Heureka heuristic as well as the MLPred heuristic explained in Algorithm 2 with a threshold of 0.35.

MCC	Standard Backtracks YES	No of AFs	Prediction Backtracks YES	No of AFs
>0.8	42,088,262	31	37,881,987	34
>0.9	90,335,877	60	16,892,061	64
1	1,287,751,506	804	0	896
Total	1,420,175,645	895	54,774,048	994

Table 7. Runtime in seconds for classifying the *kwt-large* dataset using the standard Heureka heuristic as well as the MLPred heuristic explained in Algorithm 2 with a threshold of 0.35.

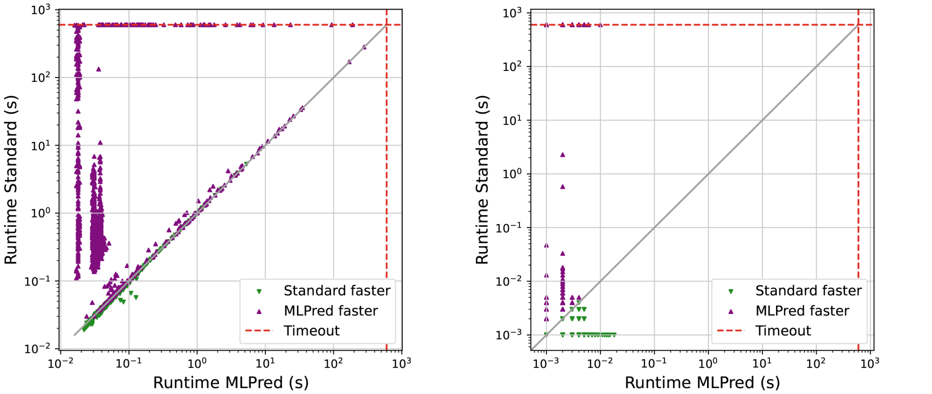
MCC	Runtime Standard	Runtime MLPred	No of AFs
> 0.8	574	552	31
> 0.9	2,148	475	60
1	30,134	3,885	804
Total	32,856	4,911	895

5.3 Evaluation and Results

The first evaluation experiment involved running Heureka on the *kwt-large* dataset using the same prediction quality threshold of 0.35. The MLPred heuristic resulted in a substantial reduction of backtracking steps required for justifying accepted arguments compared to the standard heuristic, as demonstrated in Table 6. Notably, the MLPred heuristic enabled heureka, to successfully solve 994 AFs, whereas the standard heuristic was only able to solve 895 AFs without encountering timeouts.

We also experienced a drastic decrease in runtime when using the MLPred heuristic. Table 7 shows an overview over the runtime in seconds needed to solve the 895 AFs that both heuristics were able to solve completely. The MLPred heuristic achieved a runtime reduction of 85%. The performance gain achieved by using the MLPred heuristic is also evident, when comparing the runtime for individual arguments. Figure 2a shows the runtime comparison for both heuristics. To limit the overview to instances of a certain difficulty, we only plot arguments, where the amount of backtracking steps exceeds the mean amount of backtracking steps for at least one heuristic. We can clearly see, that on the vast majority of arguments the MLPred heuristic outperformed the standard heuristic.

In order to assess the performance of the prediction heuristic on a different graph type, we ran the same experiments on the *Barabasi-test* Dataset. Again, the prediction heuristic resulted in a significant reduction in the need for backtracking. In fact, as can be seen in Table 8, the need for backtracking was eliminated almost completely.



(a) Runtime per argument for the *Kwt-large* dataset.

(b) Runtime per argument for the *Barabasi* dataset.

Fig. 2. Runtime per argument for arguments with above-average backtracking steps

We also compared the runtime for both heuristics for the *Barabasi* dataset. Interestingly, as is evident in Table 9 the standard heuristic overall was the faster choice for the AFs both heuristics could solve. This stems from the fact that Heureka needs more time to parse and build the MLPred heuristic. As the graphs in this dataset in general can be solved much faster than those in the *Kwt-large* dataset, the decreased runtime for the justification process is not enough to outweigh the overhead added by using the prediction.

However, when comparing the runtime for the individual arguments in Fig. 2b, we can see that for the hardest arguments of this dataset the MLPred heuristic performed better. Combined with the fact, that the MLPred heuristic was able to solve all AFs of this dataset we can still conclude that using a machine learning prediction was beneficial when solving the *Barabasi* dataset.

6 Limitations

Our study primarily focuses on enhancing the solution runtime for arguments classified as DC. While we successfully utilized machine learning predictions to guide the Heureka solver in justifying rejected arguments in several test cases, our approach did not yield satisfactory results when applied to a larger number of arguments. Additionally, our investigation only considered credulous acceptance under preferred semantics. Furthermore, we limited our analysis to two different graph types, namely *kwt* and *barabasi* graphs.

While *kwt* graphs are intentionally designed to pose a challenge wrt. deciding DC under preferred semantics, and *barabasi* graphs are advantageous to our study due to their tendency to contain a large number of accepted arguments, it would be beneficial to assess the efficacy of our approach on other graph types in the future. This assessment should specifically include graphs used as benchmarks

Table 8. Results for classifying the *Barabasi* dataset using the standard Heureka heuristic as well as the MLPred heuristic explained in Algorithm 2 with a threshold of 0.35.

MCC	Standard Backtracks YES	No of AFs	Prediction Backtracks YES	No of AFs
>0.5	0	4	2	4
>0.6	1,080	53	362	053
>0.7	2,521,167	467	2,843	481
>0.8	236,081,763	428	1,774	433
>0.9	1,510	29	63	29
Total	238,605,520	981	5,044	1,000

Table 9. Runtime in seconds for classifying the *Barabasi* dataset using the standard Heureka heuristic as well as the MLPred heuristic explained in Algorithm 2 with a threshold of 0.35.

MCC	Runtime Standard	Runtime MLPred	No of AFs
> 0.5	0	0	4
> 0.6	7	17	53
> 0.7	113	294	467
> 0.8	156	243	428
> 0.9	3	6	29
Total	278	561	981

in the *International Competition on Computational Models of Argumentation*[7], enabling a direct comparison to other state-of-the-art solvers.

We chose a lightweight approach, employing a standard random forest classifier trained on different graph properties. Although the classification results were reasonably good, more advanced techniques such as neural networks have demonstrated even better results and could, therefore, prove beneficial in our pursuit to improve the runtime of justification algorithms.

7 Conclusion

The goal of our research was to improve the runtime of a search-based solver, by reducing the backtracking steps needed to justify whether an argument is DC.

Our study revealed that using machine learning predictions to assist a search-based solver leads to notable advantages in minimizing backtracking steps and improving runtime in decision-making processes, specifically in the context of argument acceptance. The integration of machine learning resulted in a significant reduction of backtracking steps, achieving a minimum reduction of 96%.

[7] http://argumentationcompetition.org/index.html.

Across all datasets examined, the overall runtime could be decreased by up to 85%. Furthermore, our approach was able to enhance the solvability of argumentation frameworks within a specified time constraint.

Further research opportunities could involve combining the classifier and the solver into a standalone application, eliminating the necessity to provide the solver with external predictions. However, given that the prediction quality of the RF classifier depends on the similarity between training and testing data, exploring alternative classifiers becomes imperative. Existing studies propose that employing graph neural networks holds promise for achieving robust prediction results.

Notably, our research did not yield significant improvements for rejected arguments. Subsequent investigations should look into strategies to effectively apply predictions to rejected arguments.

Acknowledgement. The research reported here was partially supported by the Deutsche Forschungsgemeinschaft (grant 375588274).

References

1. Alviano, M.: The pyglaf argumentation reasoner. In: Technical Communications of the 33rd International Conference on Logic Programming (ICLP 2017). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik (2018)
2. Amgoud, L., Devred, C.: Argumentation frameworks as constraint satisfaction problems. In: Benferhat, S., Grant, J. (eds.) SUM 2011. LNCS (LNAI), vol. 6929, pp. 110–122. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23963-2_10
3. Biere, A., Heule, M., van Maaren, H., Walsh, T. (eds.): Handbook of Satisfiability, Frontiers in Artificial Intelligence and Applications, vol. 185. IOS Press (2009)
4. Bistarelli, S., Santini, F.: Modeling and solving AFs with a constraint-based tool: ConArg. In: Modgil, S., Oren, N., Toni, F. (eds.) TAFA 2011. LNCS (LNAI), vol. 7132, pp. 99–116. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29184-5_7
5. Cerutti, F., Gaggl, S.A., Thimm, M., Wallner, J.P.: Foundations of implementations for formal argumentation. In: Baroni, P., Gabbay, D., Giacomin, M., van der Torre, L. (eds.) Handbook of Formal Argumentation, chap. 15. College Publications (2018)
6. Craandijk, D., Bex, F.: Deep learning for abstract argumentation semantics. In: Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, pp. 1667–1673 (2020)
7. Craandijk, D., Bex, F.: Enforcement heuristics for argumentation with deep reinforcement learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, pp. 5573–5581 (2022)
8. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. **77**(2), 321–358 (1995)
9. Dvořák, W., Rapberger, A., Wallner, J.P., Woltran, S.: ASPARTIX-V19 - an answer-set programming based system for abstract argumentation. In: Herzig, A., Kontinen, J. (eds.) FoIKS 2020. LNCS, vol. 12012, pp. 79–89. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-39951-1_5

10. Dvořák, W., Dunne, P.E.: Computational problems in formal argumentation and their complexity. In: Baroni, P., Gabbay, D., Giacomin, M., van der Torre, L. (eds.) Handbook of Formal Argumentation, chap. 14. College Publications (2018)
11. Egly, U., Gaggl, S.A., Woltran, S.: Answer-set programming encodings for argumentation frameworks. Argument Comput. **1**(2), 147–177 (2010). https://doi.org/10.1080/19462166.2010.486479
12. Geilen, N., Thimm, M.: Heureka: a general heuristic backtracking solver for abstract argumentation. In: Black, E., Modgil, S., Oren, N. (eds.) TAFA 2017. LNCS (LNAI), vol. 10757, pp. 143–149. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-75553-3_10
13. Hoffmann, S.: Investigating the influence of graph properties on the prediction quality of machine learning methods in the context of abstract argumentation (2023)
14. Klein, J., Kuhlmann, I., Thimm, M.: Graph neural networks for algorithm selection in abstract argumentation. In: ArgML@ COMMA, pp. 81–95 (2022)
15. Kuhlmann, I., Thimm, M.: Using graph convolutional networks for approximate reasoning with abstract argumentation frameworks: a feasibility study. In: Ben Amor, N., Quost, B., Theobald, M. (eds.) SUM 2019. LNCS (LNAI), vol. 11940, pp. 24–37. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-35514-2_3
16. Kuhlmann, I., Wujek, T., Thimm, M.: On the impact of data selection when applying machine learning in abstract argumentation. In: Computational Models of Argument, pp. 224–235. IOS Press (2022)
17. Lagniez, J.M., Lonca, E., Mailly, J.G.: Coquiaas: a constraint-based quick abstract argumentation solver. In: 2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 928–935. IEEE (2015)
18. Malmqvist, L.: Approximate solutions to abstract argumentation problems using graph neural networks. Ph.D. thesis, University of York (2022)
19. Malmqvist, L., Yuan, T., Nightingale, P., Manandhar, S.: Determining the acceptability of abstract arguments with graph convolutional networks. In: SAFA@ COMMA, pp. 47–56 (2020)
20. Newman, M.: Networks. Oxford University Press, Oxford (2018)
21. Niskanen, A., Järvisalo, M.: μ-toksia: an efficient abstract argumentation reasoner. In: Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, pp. 800–804 (2020).https://doi.org/10.24963/kr.2020/82
22. Nofal, S., Atkinson, K., Dunne, P.E.: Looking-ahead in backtracking algorithms for abstract argumentation. Int. J. Approx. Reason. **78**, 265–282 (2016)
23. Thimm, M.: Dredd - a heuristics-guided backtracking solver with information propagation for abstract argumentation. In: The Third International Competition on Computational Models of Argumentation (ICCMA 2019) (2019)
24. Thimm, M., Cerutti, F., Vallati, M.: Fudge: a light-weight solver for abstract argumentation based on sat reductions. arXiv preprint arXiv:2109.03106 (2021)
25. Vallati, M., Cerutti, F., Giacomin, M.: Predictive models and abstract argumentation: the case of high-complexity semantics. Knowl. Eng. Rev. **34** (2019)
26. Wallner, J.P., Weissenbacher, G., Woltran, S.: Advanced SAT techniques for abstract argumentation. In: Leite, J., Son, T.C., Torroni, P., van der Torre, L., Woltran, S. (eds.) CLIMA 2013. LNCS (LNAI), vol. 8143, pp. 138–154. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40624-9_9

Ranking Transition-Based Medical Recommendations Using Assumption-Based Argumentation

Kenneth Skiba[1(✉)], Matthias Thimm[1], and Johannes P. Wallner[2]

[1] Artificial Intelligence Group, University of Hagen, Hagen, Germany
{kenneth.skiba,matthias.thimm}@fernuni-hagen.de
[2] Institute of Software Technology, TU Graz, Graz, Austria
wallner@ist.tugraz.at

Abstract. We present a general framework to rank assumption in assumption-based argumentation frameworks (ABA frameworks), relying on their relationship to other assumptions and the syntactical structure of the ABA framework. We propose a new family of semantics for ABA frameworks that is using reductions to the abstract argumentation setting and leveraging existing ranking-based semantics for abstract argumentation. We show the suitability of these semantics by investigating a case study based on medical recommendations for patients with multiple health conditions and show that the relationship of the recommendations are enough to establish a ranking between the recommendations.

Keywords: ABA · Ranking-based semantics · TMR

1 Introduction

In recent years, the use of artificial intelligence in medicine has become increasingly popular [1,3,23]. An AI system can be used to support the decision-making process of practitioners, in particular by recommending treatments for patients with specific health conditions. A particularly challenging task is finding good recommendations for patients with several different health conditions (*multimorbidities*), where the different health conditions require different treatment approaches [12,13]. In this case, treatments need to be combined, but such a combination is not always trivial. It may be that two treatments do not mix well, or worse, that they counteract each other. Therefore, an AI system needs to take into account the interaction between different treatment approaches in order to recommend the best course of action.

The *Transition-based Medical Recommendation model* (TMR) is used to represent clinical guideline recommendations and their interactions. These recommendations consist of an action and a corresponding effect on a property. The actions of two recommendations may contradict each other. However, the TMR model is constructed based on a generic database and cannot be used directly

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 202–220, 2024.
https://doi.org/10.1007/978-3-031-63536-6_12

to reason wrt. a specific patient and their health conditions. To overcome this disadvantage and reason with TMR on specific patient data, Cyras et al. [12,13] proposed to use formal argumentation [4] as the foundation for a decision-making model. Formal argumentation is concerned with the representation of arguments and their relationships. One important approach is the abstract argumentation framework (AF) by Dung [15]. This framework uses directed graphs to represent arguments as nodes and attacks between two arguments as edges between these two arguments, where the source of an edge *attacks* the target. One way of reasoning with AFs is to use *extension-based semantics*, which specify when a set of arguments is *acceptable*.

In addition to AFs, other models of rational decision-making using argumentative reasoning have been explored in the literature. One of these are the *assumption-based argumentation frameworks* (ABA frameworks) [6,7,16,26]. These are based on deductive systems over a formal language with rules. One important component of the formal language are the so-called *assumptions*, which are used as the basis for deriving further pieces of information. Similar to AFs, one reasoning method for ABA frameworks are extension-based semantics that define when a set of assumptions is *acceptable*. Abstract argumentation frameworks and ABA frameworks are closely related; the standard approach to reasoning with ABA frameworks involves deriving an AF and a translation for the other direction exists as well [14].

The classical semantics of both AFs and ABA frameworks induce a binary classification of arguments resp. assumptions: an argument or assumption is either accepted or not. This may be considered too restrictive in real-world scenarios such as the treatment recommendation scenario from above. For AFs, *ranking-based semantics* [2,8] have been introduced to overcome this limitation, where a ranking of arguments is established based on their individual strength. Thus, we can not only state that an argument is part of an acceptable set or not, but also infer that one argument is "better" than another one.

In this paper we introduce *ranking-based semantics* for the ABA setting to rank assumptions based on their strength. Using these semantics, we can state whether one assumption is stronger than another. We present a family of ranking-based semantics based on ideas for AFs. For an ABA framework, we look at the induced AF and compute a ranking over arguments, then lift the resulting ranking back to ABA, and then re-evaluate the result in the context of ABA. In addition, we look at a case study based on [28] using the TMR model to rank medical recommendations and show that the proposed ranking formalism behaves in line with other recent AI systems for finding medical recommendations.

This paper is organised as follows. We recall the necessary background information about AFs, ranking-based semantics, ABA frameworks and the TMR model in Sect. 2. In Sect. 3, we introduce ranking-based semantics for ABA frameworks and propose a family of ranking-based semantics for ABA frameworks based on ranking-based semantics for AFs. In Sect. 4, we investigate a case study based on the TMR model to show the intuitive behaviour of our proposed semantics. Related work is discussed in Sect. 5 and Sect. 7 concludes

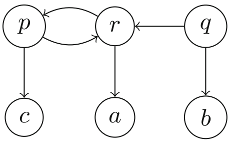


Fig. 1. Abstract argumentation framework F from Example 1.

the paper. This paper is a continuation of the workshop paper (with informal proceedings) [25] and extended with a case study in Sect. 4.

2 Preliminaries

In this section we recall the necessary preliminaries for this work. We start with *abstract argumentation framework* and their *extension-based semantics*, since they are the most basic notion we will need. After that we recall *ranking-based semantics* as an alternative the extension-based semantics. Finally we denote *assumption-based argumentation frameworks*, which uses abstract argumentation frameworks and their extension-based semantics to reason about sets of assumptions.

2.1 Abstract Argumentation Frameworks

An *abstract argumentation framework* (AF) is a directed graph $F = (Arg, Att)$ where Arg is a finite set of *arguments* and $Att \subseteq Arg \times Arg$ is an *attack relation* [15]. An argument a is said to *attack* an argument b if $(a, b) \in Att$. We say that an argument a is *defended by a set* $E \subseteq Arg$ if every argument $b \in Arg$ that attacks a is attacked by some $c \in E$. For $a \in Arg$ we define $a_F^- = \{b \mid (b, a) \in Att\}$ and $a_F^+ = \{b \mid (a, b) \in Att\}$, so the sets of attackers of a and the set of arguments attacked by a in F. For a set of arguments $E \subseteq A$ we extend these definitions to E_F^- and E_F^+ via $E_F^- = \bigcup_{a \in E} a_F^-$ and $E_F^+ = \bigcup_{a \in E} a_F^+$, respectively. If the AF is clear in the context, we will omit the index.

Example 1. Consider the argumentation framework $F = (Arg, Att)$ with

$$Arg = \{a, b, c, p, q, r\} \qquad Att = \{(r, a), (q, b), (p, c), (p, r), (r, p), (q, r)\}.$$

F is depicted as a directed graph in Fig. 1, with the nodes corresponding to arguments, and the edges corresponding to attacks.

Most semantics [5] for abstract argumentation are relying on two basic concepts: *conflict-freeness* and *admissibility*. Given $F = (Arg, Att)$, a set $E \subseteq Arg$ is

- *conflict-free* iff $\forall a, b \in E$, $(a, b) \notin Att$;
- *admissible* iff it is conflict-free, and every element of E is defended by E.

We use $cf(F)$ and $ad(F)$ for denoting the sets of conflict-free and admissible sets of an argumentation framework F, respectively. The intuition behind these concepts is that a set of arguments may be accepted only if it is internally consistent (conflict-freeness) and able to defend itself against potential threats (admissibility). The semantics proposed by Dung [15] are then defined as follows.

Definition 1. *Given $F = (Arg, Att)$, an admissible set $E \subseteq Arg$ is*

- *a* complete *extension (co) iff it contains every argument that it defends;*
- *a* preferred *extension (pr) iff it is a $\subseteq$-maximal complete extension;*
- *a* grounded *extension (gr) iff it is a $\subseteq$-minimal complete extension;*
- *a* stable *extension (stb) iff $E_F^+ = A \setminus E$.*

The sets of extensions of an argumentation framework F, for these four semantics, are denoted (respectively) $co(F)$, $pr(F)$, $gr(F)$ and $stb(F)$. Note that the grounded extension is uniquely determined [15].

2.2 Ranking-Based Semantics

While extension-based semantics can only differentiate between acceptance and non-acceptance of arguments, *ranking-based semantics* [2] allow to rank arguments based on their strength.

Definition 2. *A* ranking-based semantics *ρ is a function, which maps an argumentation framework $F = (Arg, Att)$ to a preorder*[1] *$\succeq_F^\rho$ on Arg.*

Intuitively, $a \succeq_F^\rho b$ means that a is at least as strong as b in F. We further define $a \succ_F^\rho b$ to denote $a \succeq_F^\rho b$ and $b \not\succeq_F^\rho a$ and $a \simeq_F^\rho b$ to denote $a \succeq_F^\rho b$ and $b \succeq_F^\rho a$.

An example for a ranking-based semantics is the *Burden-based semantics* [2], which is based on *burden numbers* that assess the strength of an argument in relation to the strengths of its attackers. Let $\succeq_{lex}$ be the *lexicographical preference order*, which for (possibly infinite) real-valued vectors $V = (V_1, V_2, \ldots)$ and $V' = (V'_1, V'_2, \ldots)$ is defined as $V \succ_{lex} V'$ iff $\exists i$ s.t. $V_i < V'_i$ and $\forall j < i$, $V_j = V'_j$ (and $V \simeq_{lex} V'$ iff $\forall i$, $V_i = V'_i$).

Definition 3. *Let $F = (Arg, Att)$ be an AF, $a \in Arg$, and $i \in \mathbb{N}$. The* burden number *$bur_i(a)$ for argument $a \in Arg$ in iteration i is defined as*

$$bur_i(a) := \begin{cases} 1 & \text{if } i = 0 \\ 1 + \sum_{b \in a_F^-} \frac{1}{bur_{i-1}(b)} & \text{otherwise} \end{cases}$$

Let $bur(a) = (bur_0(a), bur_1(a), bur_2(a), \ldots)$ and define the Burden-based semantics *(Bbs) ranking $\succeq_F^{Bbs}$ via $a \succeq_F^{Bbs} b$ iff $bur(a) \succeq_{lex} bur(b)$ for all $a, b \in Arg$.*

[1] A preorder is a (binary) relation that is *reflexive* and *transitive*.

Example 2. Consider again the AF F from Example 1. Argument q is unattacked, hence $bur(q) = (1, 1, 1, \ldots)$. The remaining burden numbers are

$$bur(a) = (1, 2, \frac{4}{3}, \ldots) \qquad bur(b) = (1, 2, 2, \ldots) \qquad bur(c) = (1, 2, \frac{2}{3}, \ldots)$$
$$bur(p) = (1, 2, \frac{4}{3}, \ldots) \qquad bur(r) = (1, 3, 2.5, \ldots).$$

Since a and p have the same attacker r, they receive in each step the same value. We obtain the ranking $q \succ_F^{Bbs} a \simeq_F^{Bbs} p \succ_F^{Bbs} c \succ_F^{Bbs} b \succ_F^{Bbs} r$.

2.3 Assumption-Based Argumentation Frameworks

Assumption-based Argumentation (ABA) frameworks builds on a deductive system $(\mathcal{L}, \mathcal{R})$, where $\mathcal{L}$ is a formal language and $\mathcal{R}$ a set of rules of the form $r = a_0 \leftarrow a_1, ..., a_n$ with $a_i \in \mathcal{L}$. We say that a_0 is the head of the rule $(head(r) = a_0)$ and the set $\{a_1, ..., a_n\}$ is the body $(body(r) = \{a_1, ..., a_n\})$.

Definition 4. *An* ABA framework *is a tuple* $(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-})$*, where* $(\mathcal{L}, \mathcal{R})$ *is a deductive system,* $\mathcal{A} \subseteq \mathcal{L}$ *a non-empty set of assumptions, and* $^{-} : \mathcal{A} \rightarrow \mathcal{L}$ *is a so-called contrary function.*

We focus in this work on *flat* ABA frameworks, i. e., $head(r) \notin \mathcal{A}$ for each rule $r \in \mathcal{R}$.

A sentence $s \in \mathcal{L}$ is derivable from a set of assumptions $X \subseteq \mathcal{A}$ and rules $\mathtt{R} \subseteq \mathcal{R}$, denoted by $X \vdash_{\mathtt{R}} s$, if there is a finite rooted labelled tree T with the root being labelled with s, the set of labels for the leaves of T is equal to X or $X \cup \{\top\}$, and the internal nodes are labelled with $head(r)$ according to a rule $r \in \mathtt{R}$ s.t. the children are labelled with $body(r)$ or $\top$ if the body is empty. Each assumption $x \in X$ has an associated leaf labelled with x and each rule $r \in \mathtt{R}$ has an associated node in the tree. For a tree T, we denote by $asm(T)$ the set of assumptions used to derive the conclusion denoted $cl(T)$ with rules $ru(T)$.

Similar to AFs, ABA frameworks can be used as a rational argumentation-based decision-making model. Here, a set of assumptions S *attacks* a set of assumptions $Q \subseteq \mathcal{A}$ if there is $S' \subseteq S$, $\mathtt{R} \subseteq \mathcal{R}$, s.t. $S' \vdash_R \overline{a}$ for some $a \in Q$. S is *conflict-free* if S does not attack S. S *defends* assumption s if S attacks each assumption set Q that attacks $\{s\}$.

Definition 5. *For* $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \_)$ *be an ABA framework and a conflict-free set of assumptions* $S \subseteq \mathcal{A}$*, we say* S *is*

- admissible *in* D *(*$S \in ad(D)$*) if* S *defends itself,*
- complete *in* D *(*$S \in co(D)$*) if* S *is admissible and contains every assumptions set it defends,*

- grounded *in D ($S \in gr(D)$) if S is $\subseteq$-minimally complete,*
- preferred *in D ($S \in pr(D)$) if S is $\subseteq$-maximally complete, and*
- stable *in D ($S \in st(D)$) iff S attacks every assumption $a \in \mathcal{A} \setminus S$.*

Example 3. Consider the ABA framework D with assumptions $\mathcal{A} = \{a, b, c\}$ and rules:

$$r_1 : \; r \leftarrow b, c \qquad\qquad r_2 : \; q \leftarrow \qquad\qquad r_3 : \; p \leftarrow q, a$$

with $\overline{a} = r, \overline{b} = q, \overline{c} = p$. We can, e. g., derive p from $\{a\}$ with rules r_2 and r_3 and since $p = \overline{c}$ we see that $\{a\}$ attacks $\{c\}$. Furthermore, $\{a\}$ and $\emptyset$ are admissible.

AFs and ABA frameworks are closely related [14], and we can define an AF as an instance of an ABA framework and the other way around.

Definition 6. *The associated AF $F_D = (Arg, Att)$ of an ABA framework $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-})$ is given by $Arg = \{T \mid T$ is a tree for $s \in \mathcal{L}$ with $cl(T) = s\}$ and attack relation $(T, T') \in Att$ iff there is $c \in asm(T')$ s.t. $\overline{c} = cl(T)$.*

Definition 7. *Let $F = (Arg, Att)$ be an AF. The associated ABA framework of F is $ABA(F) = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-})$ with*

$$\mathcal{A} = Arg \qquad \mathcal{L} = \mathcal{A} \cup \{a^c | a \in \mathcal{A}\} \qquad \mathcal{R} = \{b^c \leftarrow a | (a, b) \in Att\}$$

and $\overline{a} = a^c$, for all $a \in \mathcal{A}$.

It can be shown [14] that if a set of assumptions S is acceptable in the ABA framework D, then S is also acceptable in the corresponding AF F_D (in the form of conclusions of an extension).

Example 4. Continuing Example 3, we can construct the corresponding AF $F_D = (Arg, Att)$ of D, with $Arg = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{p}, \mathtt{q}, \mathtt{r}\}$ where

- $\mathtt{a}$ is a tree with $asm(\mathtt{a}) = \{a\}$, $cl(\mathtt{a}) = a$, and $ru(\mathtt{a}) = \emptyset$,
- $\mathtt{b}$ is a tree with $asm(\mathtt{b}) = \{b\}$, $cl(\mathtt{b}) = b$, and $ru(\mathtt{b}) = \emptyset$,
- $\mathtt{c}$ is a tree with $asm(\mathtt{c}) = \{c\}$, $cl(\mathtt{c}) = c$, and $ru(\mathtt{c}) = \emptyset$,
- $\mathtt{p}$ is a tree with $asm(\mathtt{p}) = \{a\}$, $cl(\mathtt{p}) = p$, and $ru(\mathtt{p}) = \{r_3\}$,
- $\mathtt{q}$ is a tree with $asm(\mathtt{q}) = \emptyset$, $cl(\mathtt{q}) = q$, and $ru(\mathtt{q}) = \{r_2\}$,
- $\mathtt{r}$ is a tree with $asm(\mathtt{r}) = \{b, c\}$, $cl(\mathtt{r}) = r$, and $ru(\mathtt{r}) = \{r_1\}$

and the attack relation $Att = \{(\mathtt{q}, \mathtt{b}), (\mathtt{q}, \mathtt{r}), (\mathtt{r}, \mathtt{a}), (\mathtt{r}, \mathtt{p}), (\mathtt{p}, \mathtt{r}), (\mathtt{p}, \mathtt{c})\}$.

The corresponding graph representation can be found in Fig. 2. So, for each derivable sentence in an ABA framework, we create an argument in the corresponding AF. We know that p is derivable from $\{a\}$ by rules r_2 and r_3, hence $\mathtt{p} \in Arg$ and additionally the attacks in the AF are representing the attacks from one set of assumptions to another set of assumptions. For example, the attack $(\mathtt{p}, \; \mathtt{r}) \in Att$ is representing the fact, that $\{a\}$ attacks $\{b\}$.

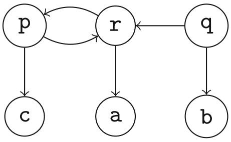


Fig. 2. Graph representation of Example 4

Note that in the following, we call argument $\mathtt{a}$, based on a tree of the form $asm(\mathtt{a}) = \{a\}$, $cl(\mathtt{a}) = a$ and $ru(\mathtt{a}) = \emptyset$, where a is an assumption, the *assumption argument* of a.

3 Ranking Assumptions

As with extension-based semantics in AFs, reasoning in ABA only distinguishes between acceptable and non-acceptable assumptions. Next we explore the applicability of ranking-based semantics for AFs to rank assumptions in ABA by defining a family of ranking-based semantics for ABA frameworks that relies on the reduction of an ABA framework to its corresponding AF, an application of a ranking-based semantics for AFs on this derived AF, and a re-interpretation of the resulting ranking over arguments in terms of assumptions. Finally, we conduct a thorough case study that illustrates the usefulness of our approach.

Definition 8. *A ranking-based semantics τ is a function that maps an ABA framework $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-})$ to a preorder $\succeq^{\tau}_{D}$ on $\mathcal{A}$.*

Intuitively, $a \succeq^{\tau}_{D} b$ means, that assumption a is at least as strong as b in D. We define the abbreviations $\succ^{\tau}_{D}$ and $\simeq^{\tau}_{D}$ as before.

We instantiate the above definition by reducing the problem in ABA to a ranking problem in AFs and utilising existing ranking-based semantics for AFs.

Definition 9. *Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-})$ be an ABA framework, $F_D = (Arg, Att)$ the corresponding AF, $a, b \in \mathcal{A}$, $\mathtt{a}$, $\mathtt{b}$ the corresponding assumption arguments, and ρ a ranking-based semantics for AFs. The ranking-based semantics ABA-ρ returns $a \succeq^{ABA\text{-}\rho}_{D} b$ iff $\mathtt{a} \succeq^{\rho}_{F_D} \mathtt{b}$.*

In other words, assumption a is at least as strong as b in D if the corresponding assumption argument $\mathtt{a}$ is at least as strong as $\mathtt{b}$ in the corresponding AF of D.

For the remainder of this paper, in particular for examples, we will use the Burden-based semantics from Definition 3 as a specific instance for a ranking-based semantics, but other existing ranking-based semantics [8] can be used instead as well.

Example 5. Consider the ABA framework D from Example 3 and its corresponding AF F_D constructed in Example 4. The ranking over arguments in F_D is then

$$\mathtt{q} \succ^{Bbs}_{F_D} \mathtt{p} \simeq^{Bbs}_{F_D} \mathtt{a} \succ^{Bbs}_{F_D} \mathtt{c} \succ^{Bbs}_{F_D} \mathtt{b} \succ^{Bbs}_{F_D} \mathtt{r}.$$

Restricting the ranking to assumption arguments gives us $\mathtt{a} \succ_{F_D}^{Bbs} \mathtt{c} \succ_{F_D}^{Bbs} \mathtt{b}$. We can project this ranking back to ABA:

$$a \succ_D^{\text{ABA-}Bbs} c \succ_D^{\text{ABA-}Bbs} b$$

Hence, a is the strongest assumption, then c, and b is the weakest assumption. The preferred extension of D is $\{a\}$, thus it is intuitive that a is the strongest assumption. While b is attacked by a fact $q \leftarrow$ meaning that b is not really strong and therefore should be ranked below c.

So the corresponding AF of an ABA framework gives us insight into the relationship between each assumption. We see that if the corresponding argument is strong or highly ranked in the corresponding AF, then the assumption will also be strong in the ABA framework. In addition, we can compare b and c with each other, which is not possible by using extension-based semantics, since both assumptions are not acceptable.

In the remainder of this section we discuss the behaviour of $\succeq^{\text{ABA-}\rho}$ in relation to the underlying ranking-based semantics ρ. If ρ behaves in a certain way, then it was shown that $\succeq^{\text{ABA-}\rho}$ satisfies proposed properties. The first property states that assumptions for which we can not derive the contrary should be ranked better than any other assumption.

Theorem 1. *If for ρ it holds that for any AF $F = (Arg, Att)$ and for all $\mathtt{a}, \mathtt{b} \in Arg$ with $\mathtt{a}_F^- = \emptyset$ and $\mathtt{b}_F^- \neq \emptyset$, $\mathtt{a} \succ_F^\rho \mathtt{b}$, then for every ABA framework $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-)$ it holds that for every assumption $a \in \mathcal{A}$ s.t. $\overline{a}$ is not derivable from any set of assumptions $Q \subseteq \mathcal{A}$ and for every assumption $b \in \mathcal{A}$ s.t. $\overline{b}$ is derivable it holds that $a \succ_D^{ABA\text{-}\rho} b$.*

Proof. Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-)$ be an ABA framework, $F_D = (A, R)$ the corresponding AF, $a, b \in \mathcal{A}$, $\mathtt{a}$, $\mathtt{b}$ the corresponding assumptions arguments, and ρ a ranking-based semantics for AFs.

Assume for ρ it holds that for any AF $F = (Arg, Att)$ and for all $a, b \in Arg$ with $a_F^- = \emptyset$ and $b_F^- \neq \emptyset$, $a \succ_F^\rho b$. Assume $\overline{a}$ is not derivable and $\overline{b}$ is derivable. Since $\overline{a}$ is not derivable, we know that $\mathtt{a}$ can not be attacked in F_D, because we do not have any argument $\mathtt{x}$ in F_D with $cl(\mathtt{x}) = \overline{a}$. Hence, $\mathtt{a}_{F_D}^- = \emptyset$. Additionally, we know that $\mathtt{b}$ is attacked at least once, because $\overline{b}$ is derivable in D, so there has to be an argument $\mathtt{x}'$ s.t. $cl(\mathtt{x}') = \overline{b}$. Hence, $\mathtt{b}_{F_D}^- \neq \emptyset$. So, we know that $\mathtt{a} \succ_{F_D}^\rho \mathtt{b}$ and therefore also $a \succ_D^{\text{ABA-}\rho} b$.

Adding attacks to an assumption, should not raise the strength of the assumption.

Theorem 2. *Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^-)$ be an ABA framework and $a \in \mathcal{A}$. Let r_{add}^- be a rule with $r_{add}^- \notin \mathcal{R}$ and $head(r_{add}^-) = \overline{a}$. D_{add}^- is a copy of D with r_{add}^- added, i. e., $D_{add}^- = (\mathcal{L}, \mathcal{R} \cup \{r_{add}^-\}, \mathcal{A}, ^-)$.*

If for ρ it holds that for any AF $F = (Arg, Att)$, it holds that for all $\mathtt{a}, \mathtt{b} \in Arg$ with $|\mathtt{a}^-| < |\mathtt{b}^-|$, $\mathtt{a} \succ^\rho_F \mathtt{b}$ and for all $\mathtt{c}, \mathtt{d} \in Arg$ either $\mathtt{c} \succeq^\rho_F \mathtt{d}$ or $\mathtt{d} \succeq^\rho_F \mathtt{c}$, then for all ABA frameworks $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \bar{\ })$ it holds for all $a, b \in \mathcal{A}$ with $a \neq b$ that $a \succeq^\tau_{D^-_{add}} b$ implies $a \succeq^{ABA\text{-}\rho}_D b$.

Proof. Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \bar{\ })$ be an flat ABA framework, $F_D = (Arg, Att)$ the corresponding AF and ρ a ranking-based semantics for AFs. Let r^-_{add} is a new rule for $a \in \mathcal{A}$, where $r^-_{add} \notin \mathcal{R}$ and $head(r^-_{add}) = \overline{a}$ and D^-_{add} is a copy of D with r^-_{add} added, i.e. $D^-_{add} = (\mathcal{L}, \mathcal{R} \cup \{r^-_{add}\}, \mathcal{A}, \bar{\ })$ and let $F_{D^-_{add}}$ be the corresponding AF.

Assume for ρ it holds that for all $\mathtt{a}, \mathtt{b} \in Arg$ with $|\mathtt{a}^-| < |\mathtt{b}^-|$, $\mathtt{a} \succ^\rho_F \mathtt{b}$ and for all $\mathtt{c}, \mathtt{d} \in Arg$ either $\mathtt{c} \succeq^\rho_F \mathtt{d}$ or $\mathtt{d} \succeq^\rho_F \mathtt{c}$. Assume $a \succeq^{\text{ABA-}\rho}_{D^-_{add}} b$ for $b \in \mathcal{A}$ and the corresponding assumption arguments $\mathtt{a}$ and $\mathtt{b}$. First, we look at the case that r^-_{add} can not be activated, so there is no tree $\mathtt{x}$ s.t. $r^-_{add} \in ru(\mathtt{x})$ meaning that, $body(r^-_{add}) \nsubseteq \mathcal{A}$ and there is no sequence of rules $(r_1, ..., r_n, r^-_{add})$ from $\mathcal{R}$ s.t. $body(r^-_{add}) \subseteq \bigcup^n_{i=1} head(r_i) \cup \mathcal{A}$. Then the addition of r^-_{add} does not change the corresponding AF, i.e. $F_D = F_{D^-_{add}}$ and therefore $a \succeq^{\text{ABA-}\rho}_{F_{D^-_{add}}} b$ implies $a \succeq^{\text{ABA-}\rho}_{F_D} b$.

Next, we look at the case, where r^-_{add} can be activated. The addition of any attack into an AF can only raise the number of attackers for an argument and can not lower the number of attackers. Similar hold for ABA frameworks, the addition and activation of a new rule does not yield to deactivation of other rules. Hence, it holds that $|\mathtt{x}^-_{F_D}| \leq |\mathtt{x}^-_{F_{D^-_{add}}}|$ for any $x \in \mathcal{A}$ and its corresponding assumption argument $\mathtt{x}$. Since $a \succeq^\rho_{D^-_{add}} b$ holds, we know that $|\mathtt{a}^-_{F_{D^-_{add}}}| \leq |\mathtt{b}^-_{F_{D^-_{add}}}|$.

If $|\mathtt{b}^-_{F_D}| = |\mathtt{b}^-_{F_{D^-_{add}}}|$, then it is clear that $|\mathtt{a}^-_{F_D}| \leq |\mathtt{b}^-_{F_D}|$ and it holds that $\mathtt{a} \succeq^\rho_{F_D} \mathtt{b}$ and therefore also $a \succeq^{\text{ABA-}\rho}_D b$.

For $|\mathtt{b}^-_{F_D}| < |\mathtt{b}^-_{F_{D^-_{add}}}|$ we know that we can derive $\overline{a}$ in $F_{D^-_{add}}$ and this activates a rule r' with $\overline{a} \in body(r')$ and this rule is needed to activate rule r'' with $head(r'') = \overline{b}$. This implies that $\overline{a}$ can not be derived in D otherwise we could activate r' in D as well and that means that $|\mathtt{b}^-_{F_D}| < |\mathtt{b}^-_{F_{D^-_{add}}}|$ could not hold. Since $\overline{a}$ can not be derived this implies $|\mathtt{a}^-_{F_D}| = 0$ and therefore $|\mathtt{a}^-_{F_D}| \leq |\mathtt{b}^-_{F_D}|$ and also $\mathtt{a} \succeq^\rho_{F_D} \mathtt{b}$, which implies $a \succeq^{\text{ABA-}\rho}_D b$.

Cyras and Toni [14] have shown that the acceptance of extension-based semantics coincides for ABA frameworks and their corresponding AFs. However, the transformation from an ABA framework to an AF and back to an ABA framework does add new rules and therefore changes the framework. However, transforming an ABA framework to an AF and back should not change the ranking.

Theorem 3. *If for ρ it holds that for any AF $F = (Arg, Att)$, it holds that for all $\mathtt{a}, \mathtt{b} \in Arg$ with $|\mathtt{a}^-| < |\mathtt{b}^-|$, $\mathtt{a} \succ^\rho_F \mathtt{b}$ and for all $\mathtt{c}, \mathtt{d} \in Arg$ either $\mathtt{c} \succeq^\rho_F \mathtt{d}$ or $\mathtt{d} \succeq^\rho_F \mathtt{c}$, then for every ABA framework $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ and F_D the corresponding AF to D, and $ABA(F_D)$ the corresponding ABA framework to F_D, it holds for any pair $a, b \in \mathcal{A}$ that we have $a \succeq^\tau_D b$ iff $a \succeq^{ABA\text{-}\rho}_{ABA(F_D)} b$.*

Proof. Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ be a flat ABA framework, $F_D = (Arg, Att)$ the corresponding AF, $ABA(F_D)$ the corresponding ABA framework of F_D, $F_{ABA(F_D)}$ the corresponding AF to $ABA(F_D)$ and ρ a ranking-based semantics for AFs. Let $a, b \in \mathcal{A}$, $\mathtt{a}$ be the corresponding assumptions argument of a and $\mathtt{b}$ be the corresponding assumption argument of b.

Assume for ρ it holds that for all $\mathtt{a}, \mathtt{b} \in Arg$ with $|\mathtt{a}^-| < |\mathtt{b}^-|$, $\mathtt{a} \succ^\rho_F \mathtt{b}$ and for all $\mathtt{c}, \mathtt{d} \in Arg$ either $\mathtt{c} \succeq^\rho_F \mathtt{d}$ or $\mathtt{d} \succeq^\rho_F \mathtt{c}$ and $a \succeq^{\text{ABA-}\rho}_D b$. If a sentence is derivable in D, then there is a corresponding argument in F_D and every argument in F_D is an assumption in $ABA(F_D)$ and since assumptions are always derivable, we know that everything, which is derivable in D is also derivable in $ABA(F_D)$. This implies that the number of attacker for any assumption argument $\mathtt{a}$ in F_D is equal to the number of attacker for the corresponding assumption argument in $F_{ABA(F_D)}$. Since ρ satisfies CP and Total and $a \succeq^{ABA-\rho}_D b$, we know $|\mathtt{a}^-_{F_D}| \leq |\mathtt{b}^-_{F_D}|$ and since the number of attacker is the same in F_D and $F_{ABA(F_D)}$, i.e. $|(a)^-_{F_D}| = |(a)^-_{F_{ABA(F_D)}}|$, we have $|\mathtt{a}^-_{F_{ABA(F_D)}}| \leq |\mathtt{b}^-_{F_{ABA(F_D)}}|$. Then $\mathtt{a} \succeq^\rho_{F_{ABA(F_D)}} b$ and therefore also $a \succeq^{\text{ABA-}\rho}_{ABA(F_D)} b$.

A self-contradicting assumption should be ranked worse than any other assumption.

Theorem 4. *If for ρ it holds that for any AF $F = (Arg, Att)$, it holds that for all $\mathtt{a}, \mathtt{b} \in Arg$ with $(\mathtt{a}, \mathtt{a}) \notin Att$ and $(\mathtt{b}, \mathtt{b}) \in Att$, $\mathtt{a} \succ^\rho_F \mathtt{b}$, then for every ABA framework $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ the following holds for every assumptions $a, b \in \mathcal{A}$, if $\{a\} \nvdash_\mathcal{R} \overline{a}$ and $\{b\} \vdash_\mathcal{R} \overline{b}$ then $a \succ^{ABA\text{-}\rho}_D b$.*

Proof. Let $D = (\mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{})$ be an ABA framework, $F_D = (Arg, Att)$ the corresponding AF, $a, b \in \mathcal{A}$, the corresponding assumptions arguments $\mathtt{a}$, $\mathtt{b}$, and ρ a ranking-based semantics for AFs.

Assume for ρ it holds that for any AF $F = (Arg, Att)$, it holds that for all $\mathtt{a}, \mathtt{b} \in Arg$ with $(\mathtt{a}, \mathtt{a}) \notin Att$ and $(\mathtt{b}, \mathtt{b}) \in Att$, $\mathtt{a} \succ^\rho_F \mathtt{b}$ and $\{a\} \nvdash_\mathcal{R} \overline{a}$ and b with $\{b\} \vdash_\mathcal{R} \overline{b}$. This implies that $(\mathtt{b}, \mathtt{b}) \in R$ and $(\mathtt{a}, \mathtt{a}) \notin R$. So, $\mathtt{b}$ attacks itself and also an assumption argument $\mathtt{x}$ for $x \in \mathcal{A}$ can only attack it self if $\{x\} \vdash_\mathcal{R} \overline{x}$, hence $\mathtt{a}$ can not attack it self. Hence, we know $\mathtt{a} \succ^\rho_{F_D} \mathtt{b}$ and this implies $a \succ^{\text{ABA-}\rho}_D b$.

4 Case Study

First, we recall the *Transition-based Medical Recommendation (TMR)* model introduced in [28] and used to construct ABA frameworks in [12,13].

The TMR model is used to represent clinical guideline recommendations for multimorbidity situations, i. e. situations where multiple health conditions need to be managed simultaneously. In addition, TMR can identify recommendations that are in conflict with each other. Using this conflict information we construct ABA frameworks as proposed in [12,13].

Definition 10. *A* recommendation *R is a tuple $R = (A, \delta, \mathcal{C})$ where:*

- *R is a* name*;*
- *A is an associated* action*;*
- *$\delta \in [-1, 1]$ is the* deontic strength*, where $\delta \geq 0$ means R recommends* performing *A and $\delta < 0$ recommends* avoiding *A;*
- *$\mathcal{C} = \langle c^1, \dots, c^n \rangle$ is a set of* contributions *with contribution c^i being a tuple $(\mathcal{P}, \mathcal{E}, v_I, v_T, o)$ with:*
  - *affected* property *$\mathcal{P}$,*
  - effect *$\mathcal{E}$ of A on $\mathcal{P}$,*
  - initial value *v_I of $\mathcal{P}$,*
  - target value *v_T of $\mathcal{P}$ after A was applied,*
  - *value $o \in \{-, \_, +\}$ of contribution indicating importance.*

We denote with R *the set of recommendations.*

Note that our definitions are a simplification of the original formal description, which can be found in [28].

Example 6. Consider recommendations

$$R_1 = (\text{Adm. NSAID}, 0.5, (\text{Blood Coag}, \text{decrease}, \text{normal}, \text{low}, +))$$
$$R_2 = (\text{Adm. Aspirin}, -0.5, (\text{Gastro. Bleeding}, \text{increase}, \text{normal}, \text{high}, -))$$

from [28]. For recommendation R_1 we have action *administering NSAID* with a strength of 0.5, which means that this action should take place, the effect of the action is to *decrease* the property *Blood Coagulation* form start value *normal* to *low*. In other words, recommendation R_1 can be translated to: *NSAID should be administered to decrease Blood Coagulation from normal to low.* while recommendation R_2 states: *Aspirin should not be administered, because it increases Gastrointestinal bleeding.*

The TMR can be used to identify conflicts or contradictions between recommendations. For example, one recommendation may suggest an action, while another recommendation urges avoiding the same action. A clinician should not follow two conflicting recommendations.

Definition 11. *A* contradiction interaction *between recommendations $R, R' \in \mathbb{R}$ is a tuple (R, R').*

The set of interactions is denoted with I.

Example 7. The two recommendations from Example 6 are contradictions to each other, since *administering NSAID* means that one should administer *Aspirin* and *Ibuprofen.* So, R_1 recommends administering Aspirin, while R_2 suggest to avoid administering Aspirin. Hence $(R_1, R_2) \in \mathsf{I}$.

The TMS recommendation and the interactions between these recommendations are for general patients. In general, however, the choice of guidelines should be based on the patient's specific medical background. Not every drug combination is suitable for every patient. A patient may be allergic to a particular drug, so that drug should not be administered. So the reasoning behind the recommendations should be based on a specific *context* or patient. We denote the *context* of a patient by S.

Next, we present a case study based on data from [28] to show that our proposed ranking over assumptions behaves intuitively in the context of medical recommendations. Similar to [12], we focus on the contradiction interactions between breast cancer (BC) and hypertension (HT) guidelines.

To construct an ABA framework based on TMS recommendations, we use the formalism of [12]. The authors defined ABA^+G frameworks, which are extensions of ABA frameworks, where additional information like a preference order over the set of assumptions as well as goals and a preoder over these goals are needed to construct an ABA^+G framework. Our approach does not need these additional information to reason and to find the best recommendations, a simple ABA framework is sufficient. Not only additional information is needed for ABA^+G but also the computational complexity of the credulous resp. sceptical acceptance problems for ABA^+G frameworks are higher than for ABA frameworks [19].[2] Recommendations are represented by assumptions, while the corresponding actions and effects are modelled by rules and the context of a patient is represented by facts.

Definition 12 **([12,13]).** *Given recommendations* R*, interactions* I *and context* S*, the* ABA patient framework *is defined via* $D_p = (\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-})$*, where:*

$A = \{R : (A, \delta, \mathcal{C}) \in \mathsf{R}\}$*, assumptions are the recommendations;*
$\mathcal{R}_a = \mathcal{R}_a^+ \cup \mathcal{R}_a^-$*, rules representing actions of recommendations, where*

$$\mathcal{R}_a^+ = \{A \leftarrow R : (A, \delta, \mathcal{C}) \in \mathsf{R}, \delta \geq 0\},$$
$$\mathcal{R}_a^- = \{not\ A \leftarrow R : (A, \delta, \mathcal{C}) \in \mathsf{R}, \delta < 0\};$$

$\mathcal{R}_e = \mathcal{R}_e^+ \cup \mathcal{R}_e^-$*, rules representing effects on properties brought about by actions, where*

[2] An assumption a is credulously (sceptically) accepted wrt. a given semantics iff it is contained in at least one (all) acceptable sets of assumptions (wrt. that semantics).

$$\mathcal{R}_e^+ = \{\mathcal{EP} \leftarrow A : (A, \delta, \mathcal{C}) \in \mathsf{R}, \delta \geq 0, (\mathcal{P}, \mathcal{E}, v_I, v_T, o) \in \mathcal{C}\},$$
$$\mathcal{R}_e^- = \{not\ \mathcal{EP} \leftarrow not\ A : (A, \delta, \mathcal{C}) \in \mathsf{R}, \delta < 0, (\mathcal{P}, \mathcal{E}, v_I, v_T, o) \in \mathcal{C}\};$$

$\mathcal{R}_s = \{v_I\mathcal{P} \leftarrow \ : v_I\mathcal{P} \in \mathcal{S}\}$, *facts representing the patient's state* $\mathcal{S}$, *where*

$$\mathcal{S} \subseteq \bigcup_{R \in \mathsf{R}} \{v_I\mathcal{P} : (\mathcal{P}, \mathcal{E}, v_I, v_T, o) \in \mathcal{C}, R = (A, \delta, \mathcal{C})\};$$

$\mathcal{R}_c = \mathcal{R}_c^+ \cup \mathcal{R}_c^-$, *rules representing contradicting interactions between recommendations, where*

$$\mathcal{R}_c^+ = \{\overline{R_j} \leftarrow R_i, int_{i,j} : (R_i, R_j, \mu) \in \mathsf{I}, \delta_i \geq 0\},$$
$$\mathcal{R}_c^- = \{\overline{R_i} \leftarrow R_j, int_{i,j}, v_{I,j}\mathcal{P}_j : (R_i, R_j, \mu) \in \mathsf{I}, (R_j, A_j, \delta_j, \mathcal{C}_j) \in \mathsf{R}, (\mathcal{P}_j, \mathcal{E}_j, v_{I,j}, v_T, -) \in \mathcal{C}_j, \delta_i < 0\};$$

$\mathcal{R} = \mathcal{R}_a \cup \mathcal{R}_e \cup \mathcal{R}_s \cup \mathcal{R}_c \cup \{int_{i,j} \leftarrow \ : (R_i, R_j) \in \mathsf{I}\}$;
By convention, $\mathcal{L}$ *and* $^-$ *are implicit from* $\mathcal{A}$ *and* $\mathcal{R}$ *as follows: unless* $\overline{x}$ *appears in either* $\mathcal{A}$ *or* $\mathcal{R}$*, it is different from the sentences appearing in* $\mathcal{A}$ *or* $\mathcal{R}$*; thus,* $\mathcal{L}$ *consists of all the sentences appearing in* $\mathcal{R}$*,* $\mathcal{A}$ *and* $\{\overline{\alpha} : \alpha \in \mathcal{A}\}$.

Example 8. Taking the recommendations of the case study from [28] focusing on the contradicting interactions between breast cancer and hypertension we get following recommendations: Let $\mathsf{R} = \{R_2, R_3, R_4, R_8\}$ with:

- $R_2 = (\text{Std. Exercise}, 0.5, \{$
$$(\text{Fatigue, decrease, high, normal}, +)$$
$$(\text{Fitness, decrease, high, normal}, +)$$
$$(\text{Pain, decrease, high, normal}, +)\}$$
- $R_3 = (\text{Low Int. Exercise}, 0.5, \{$
$$(\text{Fatigue, decrease, high, normal}, +)$$
$$(\text{Fitness, decrease, high, normal}, +)$$
$$(\text{Pain, decrease, high, normal}, +)\}$$
- $R_4 = (\text{Exercise}, -1, \{(\text{Body Temp, increase, high, very high}, -)\})$
- $R_8 = (\text{High Int. Exercise}, -0.5, \{(\text{Blood Pressure, increase}, ?, ?, -)\}$

The interactions between these recommendations are then: $\mathsf{I} = \{(R_2, R_4), (R_3, R_4), (R_2, R_8)\}$. So, recommendations R_2 and R_4 are in a conflict and should not be followed simultaneously. To model patient-orientated reasoning let us consider *Patient A* from [12]. *Patient A* has increased *Blood Pressure* and *high Body Temperature*. The corresponding ABA framework to *Patient A* is: $D_{P_a} =$

$(\mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-}) :$

$$\begin{aligned}
\mathcal{A} = &\{R_2, R_3, R_4, R_8\}\\
\mathcal{R} = &\{\text{Std. Exercise} \leftarrow R_2,\\
&\text{Low Int. Exercise} \leftarrow R_3,\\
&\textit{not}\ \text{Exercise} \leftarrow R_4,\\
&\textit{not}\ \text{High Int. Exercise} \leftarrow R_8\} \cup\\
&\{\text{increase Body Temp} \leftarrow \text{Std. Exercise},\\
&\text{decrease Fatigue} \leftarrow \text{Std. Exercise},\\
&\text{decrease Pain} \leftarrow \text{Std. Exercise},\\
&\text{decrease Fatigue} \leftarrow \text{Low Int. Exercise},\\
&\text{decrease Pain} \leftarrow \text{Low Int. Exercise},\\
&\textit{not}\ \text{increase Blood Pressure} \leftarrow \textit{not}\ \text{High Int. Exercise}\},\\
&\textit{not}\ \text{increase Body Temp} \leftarrow \textit{not}\ \text{Exercise}\} \cup\\
&\{\text{Blood Pressure} \leftarrow\ , \text{high Body Temp.} \leftarrow\ \} \cup\\
&\{\overline{R_4} \leftarrow R_2, int_{2,4},\\
&\overline{R_2} \leftarrow R_4, init_{2,4}, \text{high Body Temp.},\\
&\overline{R_4} \leftarrow R_3, init_{3,4},\\
&\overline{R_3} \leftarrow R_4, init_{3,4}, \text{high Body Temp.},\\
&\overline{R_8} \leftarrow R_2, init_{2,8},\\
&\overline{R_2} \leftarrow R_8, init_{2,8}, \text{Blood Pressure}\} \cup\\
&\{init_{2,4} \leftarrow\ , init_{3,4} \leftarrow\ , init_{2,8} \leftarrow\ \}
\end{aligned}$$

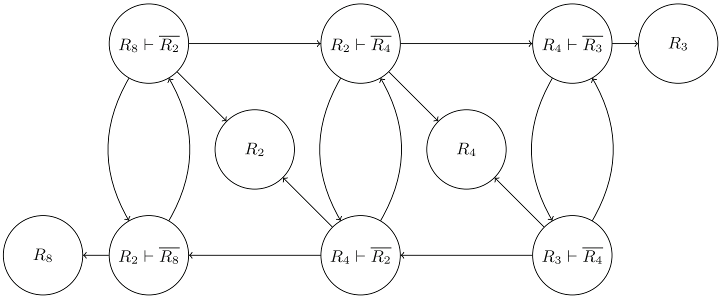

$R_8 \vdash \overline{R_2}$
$R_2 \vdash \overline{R_4}$
$R_4 \vdash \overline{R_3}$
R_3
R_2
R_4
R_8
$R_2 \vdash \overline{R_8}$
$R_4 \vdash \overline{R_2}$
$R_3 \vdash \overline{R_4}$

Fig. 3. Simplified graph representation of the case study, where only arguments with a recommendation or their contrary in the conclusion are depicted.

A simplified graph representation of the corresponding AF to D_{P_a} can be found in Fig. 3, where only arguments are depicted, which are relevant for the reasoning process, i. e., only arguments with recommendations or their contrary in their conclusion. The remaining arguments are only leaf arguments with terms like *Std. Exercise* and not relevant to the acceptance of other arguments. The preferred extensions of D_{P_a} are $\{R_3, R_8\}, \{R_4, R_8\}$ and $\{R_2, R_3\}$. These extensions do not give us any insight of which recommendation to follow, as all recommendations are credulously accepted but none of them are sceptically accepted. To identify the 'best' recommendations, we use the formalism proposed in Definition 9 and use again *Burden-based semantics* as the underlying ranking-based semantics. The resulting ranking of the four recommendations is

$$R_8 =^{ABA\text{-}Bbs}_{D_{P_a}} R_3 \succ^{ABA\text{-}Bbs}_{D_{P_a}} R_4 =^{ABA\text{-}Bbs}_{D_{P_a}} R_2$$

Recommendations R_8 and R_3 are the best recommendations to follow. These two recommendations are not in a conflict with each other and actually form a preferred extension, so we can follow both these recommendations together without any problem. Hence, for patient A we recommend to *not do High Intensive Exercise*, because this will increase their *Blood Pressure*, but the patient should do *low Intensive Exercise*, because this *decreases Fatigue, Fitness and Pain.*

The results of the case study in Example 8 are consistent with the informal discussion of [28] as well as the resulting reasoning of [12], both of which suggest following R_8 and R_3. The case study shows that the individual strength of each recommendation are already enough to reason with and we do not need additional information like a preference order over the recommendations like needed in the approach of [12]. In general recommendations with less interaction with other recommendations will be ranked highly, since in the corresponding AF these recommendations only have small number of attackers. Thinking a bit further we realise that avoiding following two contradicting recommendation is the main motivation of TMR.

5 Related Work

One of the most discussed topics in structured argumentation are preferences over uncertain information. These preferences state that information a is better or more believable than information b. A number of frameworks that work with preferences can be found in the literature such as ASPIC$^+$ [10,20–22,24], ABA$^+$ [11,14] or p_ABA [27]. While ABA$^+$ and p_ABA are extensions of ABA, ASPIC$^+$ is a general-purpose structure argumentation framework, with focus on preferences. Prakken [24] has shown that flat ABA frameworks can be instantiated as ASPIC$^+$ frameworks. In addition to an ABA framework, ABA$^+$ receives a preference over the assumptions as input. Using these preferences a new attack relation is defined. Similar to ABA$^+$, p_ABA receives a preference as input in addition to the ABA framework. However, the preference in p_ABA is over the sentences $\mathcal{L}$. In these frameworks, the preferences are preorders over rules and ordinary premises (ASPIC$^+$), assumptions (ABA$^+$) or sentences (p_ABA). Hence, these preferences are similar to our rankings over assumptions. All these preferences can be seen as a notion of strength, if an assumption a is preferred to an assumption b in an ABA$^+$ framework, then this relationship between a and b can be seen as a being better than b. However, all these frameworks receive their preferences as an input rather than calculating the preorders.

In ASPIC$^+$ and ABA$^+$ preferences are used to disable or reverse attacks. If the target of an attack is considered better than the attacker, the attack is discarded or reversed so that the attacker becomes the attacker.

Another application is to use the underlying ranking over assumptions to construct the corresponding ABA$^+$ framework for an ABA framework. So, we take an ABA framework and compute a ranking over the assumption with any ranking-based semantics like ABA-*Bbs* to then construct an ABA$^+$ framework using our ranking as a preference order. An ABA$^+$ framework constructed in such a way has similarities with the underlying ABA framework for example the conflict-free sets are the same. Thus, we can transform any ABA framework into an ABA$^+$ framework without additional information such as a preference order.

p_ABA uses preferences to discredit sets of assumptions. Wakaki [27] proposes preorders over sets of assumptions. However, their approach has two major differences: first, in p_ABA preferences are part of the input, and second, they can only distinguish sets of assumptions satisfying an extension-based semantics.

In the literature, ranking-based semantics are used to refine extension-based reasoning for AFs. For example Bonzon et al. [9] use the aggregated strength values of each argument of a set to compare two sets. Whereas Konieczny et.al. [18] compare two sets of arguments using a pairwise comparison based on a criterion like the number of arguments within the first set that are not attacked by the second set. Thus, the presented ranking-based semantics for ABA frameworks are the first step towards refining extension-based reasoning for ABA frameworks.

Heyninck et al. [17] have discussed ranking-based semantics for ABA frameworks as well, however their focus is more on the numerical strength value each assumptions receives rather than the relationship between each assumption with respect to their strength like presented in this paper.

6 Limitations

The biggest limitation of the approach discussed in this paper is the initial construction of the ABA framework based on the recommendation data given. In Example 8 we already see such limitations, even-though our case study only contains four recommendations the corresponding ABA framework has already 21 rules. Hence, with an increasing number of recommendations the corresponding ABA framework could be to big to handle.

7 Conclusion

In this paper, we discussed the problem of individual strength of assumptions in ABA frameworks. We proposed a general framework to rank assumptions based on their strength within an ABA framework without additional information such as a preference order. We also defined a family of ranking-based semantics for ABA based on approaches and ideas for AFs. For an ABA framework we construct the corresponding AF then apply known ranking-based semantics in order to rank arguments in the corresponding AF to finally re-interpret this ranking in the ABA setting. In addition, we used the proposed semantics on a case study to rank recommendations in the TMR model.

As for future work, we want to look at other structured argumentation frameworks such as ASPIC$^+$ and apply similar ideas in order to rank individual elements of the ASPIC$^+$ framework based on their strength alone. Our current approach uses AFs in order to rank assumptions. As a follow-up we want to propose direct approaches using only the ABA framework without the help of the corresponding AF.

Acknowledgements. The research reported here was supported by the Deutsche Forschungsgemeinschaft under grants 375588274 and 506604007 and by the Austrian Science Fund (FWF) P35632. For the purpose of open access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

References

1. Ahsan, M.M., Siddique, Z.: Machine learning-based heart disease diagnosis: a systematic literature review. Artif. Intell. Med. **128**, 102289 (2022)
2. Amgoud, L., Ben-Naim, J.: Ranking-based semantics for argumentation frameworks. In: Proceedings of SUM 2013, pp. 134–147 (2013)
3. Ashfaq, A., Lingman, M., Sensoy, M., Nowaczyk, S.: Deed: Deep evidential doctor. Artif. Intell. **325**, 104019 (2023)
4. Atkinson, K., et al.: Toward artificial argumentation. AI Magazine **38**(3), 25–36 (2017)
5. Baroni, P., Caminada, M., Giacomin, M.: Abstract argumentation frameworks and their semantics. In: Baroni, P., Gabbay, D., Giacomin, M., van der Torre, L. (eds.) Handbook of Formal Argumentation, pp. 159–236. College Publications (2018)
6. Bondarenko, A., Dung, P.M., Kowalski, R.A., Toni, F.: An abstract, argumentation-theoretic approach to default reasoning. Artif. Intell. **93**, 63–101 (1997)
7. Bondarenko, A., Toni, F., Kowalski, R.A.: An assumption-based framework for non-monotonic reasoning. In: Pereira, L.M., Nerode, A. (eds.) Proceedings of LPNMR, pp. 171–189. MIT Press (1993)
8. Bonzon, E., Delobelle, J., Konieczny, S., Maudet, N.: A comparative study of ranking-based semantics for abstract argumentation. In: Proceedings of AAAI 2016, pp. 914–920 (2016)
9. Bonzon, E., Delobelle, J., Konieczny, S., Maudet, N.: Combining extension-based semantics and ranking-based semantics for abstract argumentation. In: Proceedings of KR 2018, pp. 118–127 (2018)
10. Caminada, M., Amgoud, L.: On the evaluation of argumentation formalisms. Artif. Intell. **171**(5–6), 286–310 (2007)
11. Cyras, K.: ABA+: assumption-based argumentation with preferences. Ph.D. thesis, Imperial College London (2017)

12. Cyras, K., Oliveira, T.: Resolving conflicts in clinical guidelines using argumentation. In: Elkind, E., Veloso, M., Agmon, N., Taylor, M.E. (eds.) Proceedings of AAMAS 2019, pp. 1731–1739 (2019)
13. Cyras, K., Oliveira, T., Karamlou, A., Toni, F.: Assumption-based argumentation with preferences and goals for patient-centric reasoning with interacting clinical guidelines. Argument Comput. **12**(2), 149–189 (2021)
14. Cyras, K., Toni, F.: ABA+: assumption-based argumentation with preferences. In: Proceedings of KR 2016, pp. 553–556 (2016)
15. Dung, P.M.: On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning. Logic Programming and n-Person Games, Artificial Intelligence (1995)
16. Dung, P.M., Kowalski, R.A., Toni, F.: Assumption-based argumentation. In: Simari, G.R., Rahwan, I. (eds.) Argumentation in Artificial Intelligence, pp. 199–218. Springer, Boston (2009). https://doi.org/10.1007/978-0-387-98197-0_10
17. Heyninck, J., Raddaoui, B., Straßer, C.: Ranking-based argumentation semantics applied to logical argumentation. In: Proceedings of IJCAI 2023, pp. 3268–3276 (2023)
18. Konieczny, S., Marquis, P., Vesic, S.: On supported inference and extension selection in abstract argumentation frameworks. In: Proceedings of ECSQARU 2015, pp. 49–59 (2015)
19. Lehtonen, T., Wallner, J.P., Järvisalo, M.: Reasoning over assumption-based argumentation frameworks via direct answer set programming encodings. In: Proceedings of AAAI 2019, pp. 2938–2945. AAAI Press (2019)
20. Modgil, S., Prakken, H.: A general account of argumentation with preferences. Artif. Intell. **195**, 361–397 (2013)
21. Modgil, S., Prakken, H.: The $ASPIC^+$ framework for structured argumentation: a tutorial. Argument Comput. **5**(1), 31–62 (2014)
22. Modgil, S., Prakken, H.: Corrigendum to"a general account of argumentation with preferences". Artif. Intell. **263**, 107–110 (2018)
23. Nassif, A.B., Talib, M.A., Nasir, Q., Afadar, Y., Elgendy, O.: Breast cancer detection using artificial intelligence techniques: a systematic literature review. Artif. Intell. Med. **127**, 102276 (2022)
24. Prakken, H.: An abstract framework for argumentation with structured arguments. Argument Comput. **1**(2), 93–124 (2010)
25. Skiba, K., Thimm, M., Wallner, J.P.: Ranking-based semantics for assumption-based argumentation. In: Proceedings of FCR 2023. CEUR Workshop Proceedings, vol. 3500, pp. 44–52 (2023). CEUR-WS.org
26. Toni, F.: A tutorial on assumption-based argumentation. Argument Comput. **5**(1), 89–117 (2014)
27. Wakaki, T.: Assumption-based argumentation equipped with preferences. In: Proceedings of PRIMA 2014, pp. 116–132 (2014)
28. Zamborlini, V., et al.: Analyzing interactions on combining multiple clinical guidelines. Artif. Intell. Med. **81**, 78–93 (2017)

Argumentation-Based Probabilistic Causal Reasoning

Lars Bengel[1](✉), Lydia Blümel[1], Tjitze Rienstra[2], and Matthias Thimm[1]

[1] Artificial Intelligence Group, University of Hagen, Hagen, Germany
{lars.bengel,lydia.blumel,matthias.thimm}@fernuni-hagen.de

[2] Department of Advanced Computing Sciences, Maastricht University, Maastricht, The Netherlands
t.rienstra@maastrichtuniversity.nl

Abstract. We introduce an argumentation-based approach for conducting probabilistic causal reasoning. For that, we consider Pearl's causal models where causal relations are modelled via structural equations and a probability distribution over background atoms. The probability that some causal statement holds is then computed by constructing a probabilistic argumentation framework and determining its extensions. This framework can then be used to generate argumentative explanations for the (non-)acceptance of the causal statement. Furthermore, we present an argumentation-based version of the twin network method for dealing with counterfactuals. Finally, we show that our approach yields the same results for causal and counterfactual queries as Pearl's model.

Keywords: causality · argumentation · counterfactuals

1 Introduction

A recent work [17] presents a machine learning model capable of predicting the mortality within the next 24 h of the in-patients of a hospital with an accuracy of 95%. This impressive example of the recent advances in AI research is also an excellent example of the limits of machine learning approaches. While it is of course helpful to know which patients need immediate treatment to prevent them from dying, the model leaves us completely in the dark regarding the kind of treatment they need. Imagine this kind of algorithm to be used during a major incident where triage is necessary. Using this model to decide who receives treatment could do more harm than it helps, because patients that could be saved with simple and fast methods would be excluded from treatment. This is one of many potential applications for AI where an explanation of the output of the model is needed. Due to this issue, Explainable Artificial Intelligence (XAI)

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-63536-6_13.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 221–236, 2024.
https://doi.org/10.1007/978-3-031-63536-6_13

has become an important research area, which is a very productive but also challenging area of research [13].

A major contribution towards a formal theory of causality is the work on causal graphs by Pearl [14]. He models causal relationships with a double-layered formalism. On the one hand, there are the structural equations which are used to compute the value of an observable variable from a given set of values for a fixed number of unobservable background variables. On the other hand, there is a directed acyclic graph, which represents the causal dependencies between observable and background variables. A causal explanation is then formalized as a set of logical statements on causal dependencies. His approach has been widely recognized and, in particular, has been adopted in recent work on XAI [11,16].

For verifying a given causal explanation one needs a reasoning formalism which can process causal statements. We propose to use abstract argumentation frameworks as introduced by Dung in [6]. An abstract argumentation framework consists of a set of arguments—in our case causal statements—and a binary attack relation between them. An argumentation semantics is applied to this structure to determine sets of collectively acceptable arguments—so called extensions—which we use to represent consistent sets of causal statements. As a non-monotonic formalism, it can handle inconsistent input, which makes it well-suited for causal reasoning, where additional information can falsify a previously inferred causal dependency. In our approach, causal statements are interpreted as arguments in an abstract argumentation framework and the attack relation represents contradicting causal inferences. This allows us to question the reasoning process during a query. A representation of causal inferences with an argumentation framework offers an intuitive and well-researched access to all maximal consistent causal theories fitting some given facts.

We present two methods for integrating uncertainty into our causal argumentation frameworks. Our first approach makes use of default reasoning to accommodate inconsistent assumptions to reason from. This allows us to reason while staying ambiguous with regard to some background variables. We presented a preliminary discussion of this method in a recent workshop paper [1]. In the second approach we refine our causal argumentation frameworks by bringing probabilities into play. In order to represent Pearl's causal theory to the full extent with argumentation, we introduce probabilistic causal argumentation frameworks, which are based on the probabilistic argumentation frameworks by Hunter [10]. To summarise, our contributions are:

- We demonstrate how causal argumentation frameworks can be used to conduct defeasible reasoning on causal statements (Sect. 3.1), following up on our work [1].
- We introduce an enhanced version, probabilistic causal argumentation framework and show that it captures Pearl's probabiblistic causal reasoning adequately (Sect. 3.2).
- We employ probabilistic argumentation frameworks for reasoning with interventional and counterfactual statements and show they produce the same results as Pearl's three-step-method and twin model approach (Sect. 4).

Moreover, Sect. 2 introduces the necessary formal context, Sect. 5 discusses related works, and Sect. 7 concludes the paper. Proofs of technical results are omitted due to space restrictions and can be found in an online appendix.[1]

2 Preliminaries

We set $\mathcal{L}$ to be the language of propositional logic over a finite set of atoms At with the usual connectives $\{\wedge, \vee, \neg, \rightarrow, \leftrightarrow\}$ and $\vdash$ is the standard entailment operator. A *valuation* $val : \mathsf{At} \rightarrow \{true, false\}$ is an assignment of truth values to propositional variables. Our causal reasoning framework builds on a well-known form of default reasoning based on maximal consistent subsets [12]. We define a knowledge base Δ as a pair (K, A) where we assume that $K \subseteq \mathcal{L}$ is a set of *facts* and $A \subseteq \mathcal{L}$ is a set of *assumptions*. Facts are true, thus we assume that K is consistent while assumptions are statements that we are willing to assume true unless we have evidence to the contrary.

Definition 1. *Let $\Delta = (K, A)$ be a knowledge base and $\phi, \psi \in \mathcal{L}$. A set $\Sigma \subseteq A$ is a* maximal K-consistent subset of A *whenever $\Sigma \cup K$ is consistent and $\Sigma' \cup K$ is inconsistent for all $\Sigma' \subseteq A$ such that $\Sigma \subset \Sigma'$. We say that:*

- *Δ entails ψ (written $\Delta \mathrel{|\!\sim} \psi$) whenever $\Sigma \cup K \vdash \psi$ for every maximal K-consistent subset of A.*
- *ϕ Δ-entails ψ (written $\phi \mathrel{|\!\sim}_{\Delta} \psi$) whenever $(K \cup \{\phi\}, A)$ entails ψ.*

The argumentative part of our causal reasoning method relies on the notion of the *argumentation framework* (AF for short) as introduced by Dung [6].

Definition 2. *An argumentation framework is a pair* $\mathsf{AF} = (\mathsf{Arg}, \mathsf{R})$ *where* Arg *is a set of* arguments *and where* $\mathsf{R} \subseteq A \times A$ *is called the* attack relation.

We say that an argument $a \in \mathsf{Arg}$ *attacks* another argument $b \in \mathsf{Arg}$ iff we have that $(a, b) \in \mathsf{R}$. We may also use infix notation for attacks and write $a\mathsf{R}b$ for $(a, b) \in \mathsf{R}$. Given an AF, a *semantics* determines sets of jointly acceptable arguments called *extensions*. In this work, we only make use of the *stable semantics*, for other semantics see [6].

Definition 3. *Let* $\mathsf{AF} = (\mathsf{Arg}, \mathsf{R})$ *be an AF. A set $E \subseteq \mathsf{Arg}$ is:*

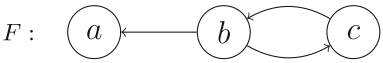
- conflict-free *iff for all $a, b \in E$ we have $(a, b) \notin \mathsf{R}$.*
- stable *iff E is conflict-free and for every $a \in \mathsf{Arg} \setminus E$ there is a $b \in E$ such that $(b, a) \in \mathsf{R}$.*

With $\mathbf{stb}(\mathsf{AF})$ we denote the set of stable extensions of an AF. For the argumentative part of our approach to reasoning with a probabilistic causal model, we use the notion of *probabilistic argumentation framework* (PAF for short) [9]. In this framework, probabilities are assigned to sets of arguments $S \subseteq \mathsf{Arg}$, called *framework states*, which implies that the existence of arguments is not independent of each other. Whenever an argument a is part of some framework state S, i.e., we have that $a \in S$, we say that a is active in S.

[1] http://mthimm.de/misc/bbrt_ratio24.pdf.

Definition 4. *A* probabilistic argumentation framework *is a pair* $\mathsf{PAF} = (\mathsf{AF}, P_{\mathsf{AF}})$ *where* $\mathsf{AF} = (\mathsf{Arg}, \mathsf{R})$ *is an argumentation framework and* $P_{\mathsf{AF}} : 2^{\mathsf{Arg}} \rightarrow [0,1]$ *is a function with* $\sum_{S \in 2^{\mathsf{Arg}}} P_{\mathsf{AF}}(S) = 1$.

Example 1. Consider the PAF in Fig. 1. We evaluate the framework by considering the different framework states and their respective extensions. For instance, the framework state $S_1 = \{a, b\}$ has a probability of 0.4 and only one stable extension $\{b\}$. On the other hand, the framework state $S_3 = \{a, b, c\}$ with probability 0.2 has two stable extensions $\{a, c\}$ and $\{b\}$.



$P_{\mathsf{AF}}(S)$	a	b	c	$\mathbf{stb}(S)$
0.4	✓	✓		$\{\{b\}\}$
0.4		✓	✓	$\{\{b\}, \{c\}\}$
0.2	✓	✓	✓	$\{\{a, c\}, \{b\}\}$

Fig. 1. The PAF $(\mathrm{F}, P_{\mathsf{AF}})$ with three frameworks states as depicted in the table.

3 Causal Reasoning

In the following, we will introduce an argumentation-based approach to perform reasoning with a causal model. The main advantage of this approach is the ability to not only determine whether some causal statement holds, but also provide an argumentative explanation on why it holds or not.

In Sect. 3.1, we introduce our approach for qualitative causal reasoning from [1], based on a modified version of Pearl's causal model [14], where we only consider Boolean-valued variables. In this scenario, we model the uncertainty via defeasibility which allows us to qualitatively answer queries directly in an argumentation framework. On the other hand, quantitative causal reasoning means computing the exact probability that the conclusion holds under the given observation. For this type of reasoning, we consider probabilistic causal models [14] and define a novel approach for answering queries with the help of a probabilistic argumentation framework (Sect. 3.2).

3.1 Defeasible Causal Reasoning

To model defeasible causal reasoning, we essentially use the causal model of Pearl [14] except that we restrict our attention to Boolean-valued variables. As described in Definition 5 below, a causal model[2] K is a set of formulas which we call *Boolean structural equations* (terminology adopted from [2]). We distinguish between two types of atoms in these equations: the *background* atoms $U(K)$ and *explainable* atoms $V(K)$. Variables that are determined outside of the model are

[2] Here, we deviate from Pearl's notation for causal models which are defined as the triple (U, V, K), explicitly listing background and explainable atoms [14]. However, with $(U(K), V(K), K)$ we recover Pearls notation of a causal model.

represented as background atoms $u \in U(K)$ and are considered unobservable and uncontrollable. An explainable atom $v \in V(K)$ is functionally dependent on other atoms of the model. We specify this dependency in the form of Boolean structural equations of the form $v \leftrightarrow \phi$, where ϕ is a logical formula over the set of atoms that v is dependent on. Intuitively, a structural equation for some explainable atom v represents the causal mechanism by which v is determined by the other atoms in the model. We use bi-implication because the represented causal mechanism determines not only when v is true, but also when v is false.

Definition 5. *A Boolean structural equation for v is a formula of the form $v \leftrightarrow \phi$ where ϕ is a propositional formula that does not contain v. A* causal model *K is a set of* Boolean structural equations*, exactly one equation κ_v for each atom $v \in V(K)$. With $U(K)$ we denote the set of background atoms appearing in K and with $V(K)$ we denote the set of explainable atoms appearing in K.*

Furthermore, a causal model induces a *causal graph G* whose vertices are the explainable atoms of the model [14]. Background atoms of the model are represented as a different type of vertex. Given a Boolean structural equation $v \leftrightarrow \phi$, we call an atom appearing in ϕ a *parent* of v. The causal graph G contains an edge from atom $v \in U \cup V$ to atom $v' \in V$ whenever v is a parent of v'. We say a causal model K is Semi-Markovian if the causal graph induced is acyclic [14].

Example 2. Suppose we are building a causal model to investigate the cause of a surfer's death by drowning at the beach. The explainable variables in this case could be $V_{surf}(K_{surf}) = \{\mathit{drowning}, \mathit{cramp}, \mathit{submersion}, \mathit{broken\text{-}board}\}$, i.e., the fact itself, two physical conditions leading to it, as well as a side-effect. The background conditions potentially leading to these variables being true are $U_{surf}(K_{surf}) = \{\mathit{jellyfish}, \mathit{strong\text{-}current}, \mathit{giant\text{-}wave}\}$. We equip these with the structural equations K_{surf}

$$\begin{aligned}
\kappa_d: &\quad \mathit{drowning} \leftrightarrow \mathit{cramp} \vee \mathit{submersion}\\
\kappa_c: &\quad \mathit{cramp} \leftrightarrow \mathit{strong\text{-}current} \vee \mathit{jellyfish}\\
\kappa_s: &\quad \mathit{submersion} \leftrightarrow \mathit{giant\text{-}wave} \wedge \mathit{strong\text{-}current}\\
\kappa_{bb}: &\quad \mathit{broken\text{-}board} \leftrightarrow \mathit{giant\text{-}wave}
\end{aligned}$$

Figure 2 depicts the causal graph for this model. The background atoms of the model are drawn using dotted lines.

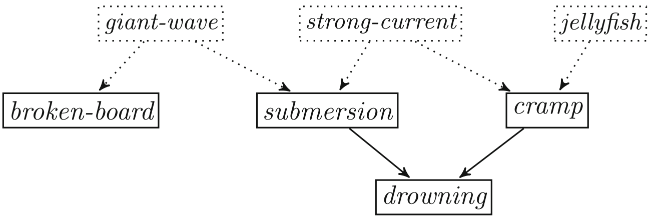


Fig. 2. Causal graph for Example 2.

We now define a *causal knowledge base* as a knowledge base, where the set of facts K is a causal model and the set of assumptions A is limited to assumptions about the background atoms in K.

Definition 6. *A* causal knowledge base *is a knowledge base* $\Delta = (K, A)$ *where* K *is a causal model and where* A *is a set of* background assumptions*, at least one for each background atom. A* background assumption *for an atom* u *is a literal* $l \in \{u, \neg u\}$*. We denote by* $\bar{l}$ *the assumption of the opposite, i. e.,* $\overline{u} = \neg u$ *and* $\overline{\neg u} = u$*.*

Since the background variables are supposed to be independent, we restrict the background assumptions to be literals. This allows us to express three possible stances towards a background atom u: we can assume just u, just $\neg u$, or both. Assuming only u ($\neg u$) amounts to assuming that u is true (false), unless we have evidence to the contrary. On the other hand, if we assume both u and $\neg u$, this represents a state of uncertainty where we are willing to consider u to be true as well as false, depending on the evidence.

Example 3. To continue Example 2 we can now construct a causal KB $\Delta = (K_{surf}, A)$ by combining the causal model K_{surf} with the set of assumptions $A = \{$*jellyfish*, *strong-current*, $\neg$*strong-current*, *giant-wave*$\}$. Intuitively, this expresses that we assume a giant wave has happened and that there are dangerous jellyfish present, but are uncertain whether there is a strong current in the area.

Given a causal knowledge base $\Delta = (K, A)$, then Δ-*entailment* can be understood as the relation between observations and predictions, i.e., an observation ϕ Δ-entails some prediction ψ, denoted by $\phi \mid\!\sim_{\Delta} \psi$, if the underlying causal model together with the observation ϕ entails the conclusion ψ. These predictions include causes as well as effects of the observation in accordance with the causal model K and the background assumptions A.

We now describe how we can transform a causal knowledge base into an argumentation framework and how to compute the Δ-entailment in that framework. For that, we adopt the approach by Cayrol et al. [4] to define an argument induced by a knowledge base $\Delta = (K, A)$. An induced argument is a pair (Φ, ψ) where $\Phi \subseteq A$ is a minimal set of assumptions (called the *premises* of the argument) that, together with K, consistently entails some *conclusion* ψ. The attacks between the arguments are given by the undercut relation. We say that an argument *undercuts* another if the conclusion of the former is the negation of a premise of the latter.

Definition 7. *Let* $\Delta = (K, A)$ *be a causal knowledge base. We define the* AF induced by Δ*, denoted with* $F(\Delta) = (\mathsf{Arg}_{\Delta}, \mathsf{R}_{\Delta})$ *as follows*

- *The set of* Δ-induced arguments Arg_{Δ} *is defined as all arguments of the form* (Φ, ψ) *such that* $\psi \in \{u, \neg u \mid U(K) \cup V(K)\}$ *and*
  - $\Phi \subseteq A$,
  - $\Phi \cup K \nvdash \bot$,

- *$\Phi \cup K \vdash \psi$, and if $\Psi \subset \Phi$ then $\Psi \cup K \nvdash \psi$.*

– *$(\Phi, \psi)\mathsf{R}_{\Delta}(\Phi', \psi')$, iff for some $\phi' \in \Phi'$ we have $\overline{\phi'} = \psi$.*

As shown by Cayrol et al. [4], there is a one-to-one correspondence between the maximal K-consistent subsets of a knowledge base and the stable extensions of an AF induced according to Definition 7. Given a causal knowledge base $\Delta = (K, A)$, this allows us to answer the question of whether ϕ Δ-entails ψ by constructing the AF induced by $(K \cup \{\phi\}, A)$ and determining whether every stable extension contains at least one argument which concludes ψ.

Proposition 1. *Let $\Delta = (K, A)$ be a causal knowledge base. Then $\phi \mid\!\sim_{\Delta} \psi$ if and only if every stable extension E of $F(K \cup \{\phi\}, A)$ contains an argument with conclusion ψ.*

Example 4. We continue with the causal knowledge base $\Delta = (K_{surf}, A)$ from Example 3. Consider the question whether observing that the surfer has drowned entails that the drowning has been caused by submersion, i.e., consider the statment whether *drowning* $\mid\!\sim_{\Delta}$ *submersion*. Submersion and a cramp are the two possible causes of drowning. It depends on the background atoms which one was the actual cause of drowning. We determine the question and the explanation via the induced AF $F = F((K \cup \{\mathit{drowning}\}, A))$, shown in Fig. 3 (we only depict arguments relevant to the conclusion of submersion). The two stable extensions of this AF are $\{a_1, a_3\}$ and $\{a_2, a_4, a_5\}$. The argument a_4 concludes *submersion*, but is only included in one of the stable extensions. Thus, *drowning* does not entail *submersion*, given the background assumptions A.

Moreover, note that the statement *drowning* $\mid\!\sim_{\Delta}$ $\neg$*submersion* does also not hold.

To conclude, we can say if we observe *drowning*, then *submersion* is a possible cause, but not necessary. The explanation for either case is then given by the corresponding stable extension containing the conclusion.

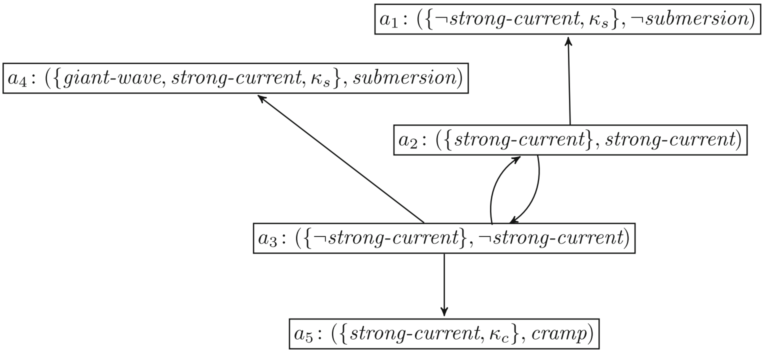

a_1: $(\{\neg\mathit{strong\text{-}current}, \kappa_s\}, \neg\mathit{submersion})$
a_4: $(\{\mathit{giant\text{-}wave}, \mathit{strong\text{-}current}, \kappa_s\}, \mathit{submersion})$
a_2: $(\{\mathit{strong\text{-}current}\}, \mathit{strong\text{-}current})$
a_3: $(\{\neg\mathit{strong\text{-}current}\}, \neg\mathit{strong\text{-}current})$
a_5: $(\{\mathit{strong\text{-}current}, \kappa_c\}, \mathit{cramp})$

Fig. 3. The AF $F(K \cup \{\mathit{drowning}\}, A)$ from Example 4.

3.2 Probabilistic Causal Reasoning

A *probabilistic causal model* [14] is defined as a causal model together with a probability assignment to every background atom. For some causal statement $\phi \mid\!\sim_{\Delta} \psi$, this allows us to determine exactly the probability that ψ holds given ϕ. As implied by Definition 8, we assume that the probabilities of the background atoms are independent, thus the causal model is considered *Markovian.*

Definition 8. *A* probabilistic causal model *is a pair* $\mathcal{C} = (K, \mathsf{P})$ *where* K *is a causal model and* $\mathsf{P} : U \to [0, 1]$ *is a probability assignment.*

Let $\mathcal{C} = (K, \mathsf{P})$ be a probabilistic causal model. A *causal state* $C \in 2^{U(K)}$ is essentially a specific configuration of the background atoms. So, if $u \in C$, then u is considered true in the state C, and otherwise u is false. We then define the *probability distribution* $P_{\mathcal{C}}$ over causal states (which correspond directly to the valuations of $U(K)$) as follows

$$P_{\mathcal{C}}(C) = \prod_{u \in C} \mathsf{P}(u) \prod_{u \in U \setminus C} (1 - \mathsf{P}(u)). \tag{1}$$

Note that the above defined function is indeed well-defined.

Proposition 2. *For any causal model* $\mathcal{C} = (K, \mathsf{P})$*, the probability distribution* $P_{\mathcal{C}}$ *sums up to* 1*.*

Example 5. Consider again the causal model K introduced in Example 2. The background atoms of K are *giant-wave* (g), *strong-current* (s) and *jellyfish* (j). We define the probability assignment P to the background atoms as follows: $\mathsf{P}(\textit{giant-wave}) = 0.8$, $\mathsf{P}(\textit{strong-current}) = 0.5$ and $\mathsf{P}(\textit{jellyfish}) = 0.2$. Then, for the probabilistic causal model $\mathcal{C} = (K, \mathsf{P})$ we compute the probability distribution of the causal states via Eq. (1) as follows: $P_{\mathcal{C}}(gs\overline{j}) = P_{\mathcal{C}}(g\overline{sj}) = 0.32$, $P_{\mathcal{C}}(gsj) = P_{\mathcal{C}}(g\overline{s}j) = P_{\mathcal{C}}(\overline{g}s\overline{j}) = P_{\mathcal{C}}(\overline{gsj}) = 0.08$ and $P_{\mathcal{C}}(\overline{g}sj) = P_{\mathcal{C}}(\overline{gs}j) = 0.02$.

For a causal statement $\phi \mid\!\sim_{\mathcal{C}} \psi$ the probability that ψ is predicted to be true, given the observation ϕ is given as the conditional probability $P_{\mathcal{C}}(\psi \mid \phi)$ [14].

Example 6. Consider the causal statement *drowning* $\mid\!\sim_{\mathcal{C}}$ *submersion.* We compute the probability $P_{\mathcal{C}}(\textit{submersion}|\textit{drowning})$ (i. e., probability of submersion given that we observe drowning) using the standard causal model approach. Continuing Example 5, we construct the probability distribution over all valuations of the background atoms, and including all the explainable atoms, whose values are determined by the background atoms, see Table 1. Computing queries based on observations simply amounts to computing a conditional probability based on the probability distribution given above. Using the definition of conditional probability we get $P_{\mathcal{C}}(\textit{submersion}|\textit{drowning}) = P_{\mathcal{C}}(\textit{submersion} \wedge \textit{drowning})/P_{\mathcal{C}}(\textit{drowning}) = 0.4/0.6 = 2/3$. Thus, the probability of submersion given that we observe drowning is 2/3.

Table 1. Partial probability distribution $P_{\mathcal{C}}$ from Example 6.

gsj	*broken-board*	*submersion*	*cramp*	*drowning*	Prob
000	0	0	0	0	0.08
001	0	0	1	1	0.02
010	0	0	1	1	0.08
011	0	0	1	1	0.02
100	1	0	0	0	0.32
101	1	0	1	1	0.08
110	1	1	1	1	0.32
111	1	1	1	1	0.08

In order to determine the probability of a statement $\phi \mathrel{|\!\!\sim}_{\mathcal{C}} \psi$, we induce a probabilistic argumentation framework PAF from the probabilistic causal model $\mathcal{C}$. For that we denote with $\mathsf{C}(\phi)$ the set of causal states in which the observation ϕ is true, defined as

$$\mathsf{C}(\phi) = \{C \in 2^{U(K)} \mid K \cup C \cup \{\neg u \mid u \notin C\} \vdash \phi\}.$$

Similar to before, an induced argument is a pair (Φ, ψ) consisting of a set of premises Φ and a conclusion ψ. The set of premises $\Phi \subseteq \{u, \neg u \mid u \in U(K)\}$ must be consistent with some causal state $C \in \mathsf{C}(\phi)$, i.e., the union of C and Φ is not contradictory, and is has to be the minimal K-consistent set to entail the conclusion ψ. The attacks of PAF are again given by the undercut relation.

We define $\mathsf{Arg}_{\mathcal{C}}(C)$ as the set of arguments consistent with a causal state $C \in 2^U$, i.e., $\mathsf{Arg}_{\mathcal{C}}(C) = \{(\Phi, \psi) \in \mathsf{Arg}_{\mathcal{C}} \mid \Phi \cup C \cup \{\neg u \mid u \notin C\} \nvdash \bot\}$, where $\mathsf{Arg}_{\mathcal{C}}$ is the set of induced arguments (see Definition 9). With that, the probability of a framework state S of the PAF is defined as the sum over the probabilities of all causal states C which are consistent with all arguments that are active in S.

Definition 9. *Let $\mathcal{C} = (K, \mathsf{P})$ be a probabilistic causal model. We define the* PAF induced by $\mathcal{C}$, *given the observation ϕ, denoted with $\mathsf{PAF}_{\mathcal{C}} = (F(\mathcal{C}), P_{\mathsf{AF}})$ with $F(\mathcal{C}) = (\mathsf{Arg}_{\mathcal{C}}, \mathsf{R}_{\mathcal{C}})$ as follows:*

- *The set of* $\mathcal{C}$-induced arguments $\mathsf{Arg}_{\mathcal{C}}$ *consists of all arguments (Φ, ψ), with $\Phi \subseteq \{u, \neg u \mid u \in U(K)\}$, such that*
  - *$\Phi \cup C \nvdash \bot$ for some $C \in \mathsf{C}(\phi)$,*
  - *$\Phi \cup K \nvdash \bot$,*
  - *$\Phi \cup K \vdash \psi$, and if $\Psi \subset \Phi$ then $\Psi \cup K \nvdash \psi$.*
- *The set of* $\mathcal{C}$-induced attacks $\mathsf{R}_{\mathcal{C}}$ *is defined via the undercut relation, i.e., an argument (Φ, ψ)* undercuts *an argument (Φ', ψ') iff for some $\phi' \in \Phi'$ we have $\phi' \equiv \overline{\psi}$.*

The probability distribution $P_{\mathsf{AF}} : 2^{\mathsf{Arg}} \to [0,1]$ *over framework states is given as*

$$P_{\mathsf{AF}}(S) = \sum_{C \in \mathsf{C}(\phi,S)} P_{\mathcal{C}}(C).$$

$$\textit{where } \mathsf{C}(\phi, S) = \{C \in \mathsf{C}(\phi) \mid S = \mathsf{Arg}_{\mathcal{C}}(C)\}.$$

Note that the above defined probability distribution P_{AF} is indeed well-defined.

Proposition 3. *For any causal model* $\mathcal{C} = (K, \mathsf{P})$ *and observation* ϕ*, the probability distribution* P_{AF} *sums up to* 1*.*

Example 7. We continue Example 5. To determine the probability of *drowning* $\mathrel{|\!\!\sim}_{\mathcal{C}}$ *submersion*, we construct the induced probabilistic argumentation framework $\mathsf{PAF}_{\mathcal{C}} = (F(\mathcal{C}), P_{AF})$, shown in Fig. 4 (only arguments relevant to the query are depicted). The framework states with non-zero probability are described in Table 2. Each framework state corresponds to one or more causal state and consists of a subset of arguments for which we can determine whether all stable extensions conclude *submersion*. In this case, only the first framework state satisfies this.

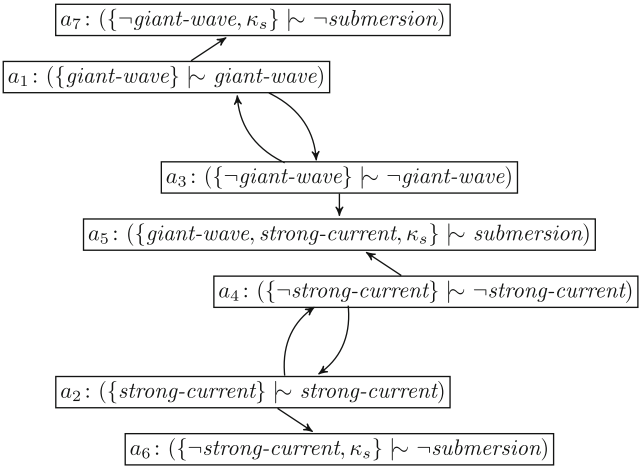

a_7: ({¬*giant-wave*, κ_s} $\mathrel{|\!\!\sim}$ ¬*submersion*)
a_1: ({*giant-wave*} $\mathrel{|\!\!\sim}$ *giant-wave*)
a_3: ({¬*giant-wave*} $\mathrel{|\!\!\sim}$ ¬*giant-wave*)
a_5: ({*giant-wave*, *strong-current*, κ_s} $\mathrel{|\!\!\sim}$ *submersion*)
a_4: ({¬*strong-current*} $\mathrel{|\!\!\sim}$ ¬*strong-current*)
a_2: ({*strong-current*} $\mathrel{|\!\!\sim}$ *strong-current*)
a_6: ({¬*strong-current*, κ_s} $\mathrel{|\!\!\sim}$ ¬*submersion*)

Fig. 4. The AF $F(K \cup \{\textit{drowning}\})$ from Example 7.

Table 2. The framework states of the induced $\mathsf{PAF}_{\mathcal{C}} = (F(K \cup \{\textit{drowning}\}), P_{\mathsf{AF}})$ which correspond to some $C \in \mathsf{C}(\phi)$.

$\mathsf{C}(\phi, S)$	$P_{\mathsf{AF}}(S)$	a_1	a_2	a_3	a_4	a_5	a_6	a_7	$S \vdash \phi$
gsj, $gs\overline{j}$	0.4	✓	✓			✓			yes
$g\overline{s}j$	0.08	✓			✓		✓		no
$\overline{g}sj$,$\overline{g}s\overline{j}$	0.1		✓	✓				✓	no
$\overline{gs}j$	0.02			✓	✓		✓	✓	no

Let $\mathcal{C} = (K, \mathsf{P})$ be a probabilistic causal model and consider some causal statement $\phi \mathrel{|\!\sim}_{\mathcal{C}} \psi$.

We can compute the probability that ψ holds given that ϕ is true via the induced probabilistic argumentation framework $\mathsf{PAF}_{\mathcal{C}} = (F(K \cup \{\phi\}), P_{\mathsf{AF}})$ as follows. With $\mathsf{S}_{[\psi=true]}$ we denote the set of framework states which entail the conclusion ψ, i.e., for which every stable extension of $\mathsf{PAF}_{\mathcal{C}}$ contains at least one argument with the conclusion ψ. In Pearl's standard causal model approach, the probability $P(\phi \mathrel{|\!\sim}_{\mathcal{C}} \psi)$ is computed as the conditional probability $P_{\mathcal{C}}(\psi|\phi) = P_{\mathcal{C}}(\psi \wedge \phi)/P_{\mathcal{C}}(\phi)$. Analogously, in our framework the probability $P_{\mathcal{C}}(\psi \wedge \phi)$ amounts to the sum of probabilities over all framework states S that entail ψ, while the probability $P_{\mathcal{C}}(\phi)$ is the sum of probabilities over all causal states in which ϕ is true. Thus, the probability $P(\phi \mathrel{|\!\sim}_{\mathcal{C}} \psi)$ is then computed as

$$P(\phi \mathrel{|\!\sim}_{\mathcal{C}} \psi) = \frac{\sum\limits_{S \in \mathsf{S}_{[\psi=true]}} P_{\mathsf{AF}}(S)}{\sum\limits_{C \in \mathsf{C}(\phi)} P_{\mathcal{C}}(C)}. \tag{2}$$

Our main theorem below states that probabilistic argumentative reasoning amounts to the same results as Pearl's classical approach, with the added value of representing causal inference through argumentative reasoning.

Theorem 1. *Let $\mathcal{C} = (K, \mathsf{P})$ be a probabilistic causal model and $\phi \mathrel{|\!\sim}_{\mathcal{C}} \psi$ is a causal statement. Then $P(\phi \mathrel{|\!\sim}_{\mathcal{C}} \psi) = P_{\mathcal{C}}(\psi|\phi)$.*

In addition to the probability that the statement is true, the induced PAF also allows us to provide different types of explanations. We can, for example, provide an explanation for the most likely scenario under which the query holds. The same can be done for the situation under which the contrary is most likely to be true. Furthermore, we might also provide an explanation for the scenario in which both outcomes are possible.

Example 8. We continue Example 7. The probability of *drowning* $\mathrel{|\!\sim}_{\mathcal{C}}$ *submersion* can then be computed via (2). Considering the framework states in Table 2, only one framework state with probability 0.4, corresponding to the causal states gsj and $gs\bar{j}$, entails the conclusion *submersion*. The sum of probabilities over the causal states that are consistent with drowning $\mathsf{C}(\textit{drowning})$ is 0.6. Thus the probability of submersion given that we observe drowning is $P(\textit{drowning} \mathrel{|\!\sim}_{\mathcal{C}} \textit{submersion}) = 0.4/0.6 = 0.\overline{66}$. In terms of explainability, we have different angles to give an explanation based on the argumentation framework. A positive explanation would be that a giant wave and a strong current cause submersion. On the other hand, we can also say that the most likely reason against submersion is that there is no strong current which means no risk of submersion, as implied by the second framework state.

4 Counterfactual Reasoning

We consider first the *interventional statements* of the form

$$\text{if } v \text{ would be } x \text{ then } \psi \text{ would be true.} \tag{3}$$

The left side of an interventional statement consists of an *action* where the atom v is intervened on, i.e., we set v to the truth value x. It is important to note that this is different from simply observing v or $\neg v$. Performing the action of setting v to x means overriding the causal mechanism that usually determines v. For some causal model K, we denote with $K_{[v=x]}$ the causal model where the structural equation of κ_v is replaced with $v \leftrightarrow x$.

Definition 10. *Let K be a causal model, let $v \in V$ be an explainable atom, and let $x \in \{\top, \bot\}$. We denote by $K_{[v=x]}$ the causal model defined by*

$$K_{[v=x]} = \{(v' \leftrightarrow \phi) \in K \mid v' \neq v\} \cup \{(v \leftrightarrow x)\}.$$

Note that we perform the intervention on the causal model K itself, which means we can apply this intervention both to a causal knowledge base $\Delta = (K, A)$ as well as a probabilistic causal model $\mathcal{C} = (K, \mathsf{P})$, depending on whether we want to reason qualitatively or quantitatively. We will then also write $\Delta_{[v=x]}$ and $\mathcal{C}_{[v=x]}$ as a shortcut for $\Delta = (K_{[v=X]}, A)$ or $\mathcal{C} = (K_{[v=x]}, \mathsf{P})$ respectively.

A *counterfactual statement* is of the form

$$\text{given } \phi,\ \text{if } v \text{ had been } x \text{ then } \psi \text{ would be true.} \tag{4}$$

Intuitively this means, if we observe ϕ and if v would have been x, then ψ would have been true. So we reason about a hypothetical or alternative scenario.

In [14], Pearl introduced two approaches to deal with counterfactual statements: a three-step procedure and the twin network method. We base our approach to counterfactual reasoning on the twin network approach. The general idea is to construct a *twin model* which consists of the actual causal model, representing the actual world, and a second model that represents the counterfactual world. Both of these worlds share the same background atoms, i.e., we have $U(K) = U(K^*)$, while for all explainable atoms $v \in V(K)$ we introduce a "counterfactual copy" $v^* \in V(K^*)$ in the counterfactual world.

Definition 11. *The* twin model *for a causal model K is the causal model K^* defined by*

$$K^* = K \cup \{(v^* \leftrightarrow \phi^*) \mid (v \leftrightarrow \phi) \in K\}.$$

Like for the intervention, we may also write Δ^* and $\mathcal{C}^*$ as a shortcut for $\Delta = (K^*, A)$ or $\mathcal{C} = (K^*, \mathsf{P})$ respectively.

First, consider the three-step procedure for evaluating counterfactual statements in a probabilistic causal model as described by Pearl [14].

Definition 12. *Given a probabilistic causal model $\mathcal{C} = (K, \mathsf{P})$, the truth of a counterfactual statement*

$$\textit{given } \phi,\ \textit{if } v \textit{ had been } x \textit{ then } \psi \textit{ would be true}$$

is determined by:

- *Step 1 (abduction) Update $P_{\mathcal{C}}$ by the evidence ϕ to obtain $P_{\mathcal{C}}(u \mid \phi)$.*

- *Step 2 (action) Modify K by the action $v = x$ to obtain $K_{[v=x]}$.*
- *Step 3 (prediction) Use the modified model $(K_{[v=x]}, P_{\mathcal{C}}(u \mid \phi))$ to compute the probability of ψ, i.e., $P_{\mathcal{C}}(\psi \mid \phi)$.*

The problem of this procedure lies in the abduction step, where we have to compute a probability distribution over configurations of the background atoms. This can be avoided by using the twin network method.

Consider a probabilistic causal model $\mathcal{C} = (K, \mathsf{P})$ and a counterfactual statement (4). Our argumentation-based approach consists of the following steps:

1. Compute the twin model $\mathcal{C}^* \cup \{\phi\}$ which includes the observation ϕ,
2. Perform the intervention $v^*{=}x$ on the counterfactual copy of v to obtain $\mathcal{C}^*_{[v^*=x]} \cup \{\phi\}$,
3. Construct the induced probabilistic AF $\mathsf{PAF}_{\mathcal{C}} = (F(\mathcal{C}^*_{[v^*=x]} \cup \{\phi\}), P_{\mathsf{AF}})$,
4. Determine the probability that ψ^* is true.

Note that the second and fourth step, representing action and prediction step of the standard three-step procedure, take place in the counterfactual world. For the third step we induce the probabilistic argumentation framework from $\mathcal{C}$ as described in Definition 9. The probability that ψ would have been true given ϕ, under the assumption that $v = x$, is calculated as the sum over the probabilities of all framework states $S \in \mathsf{S}_{[\psi^*=true]}$ for which every stable extension of the induced probabilistic argumentation framework of the twin model $\mathsf{PAF}_{\mathcal{C}} = (F((\mathcal{C} \cup \{\phi\})), P_{\mathsf{AF}})$ contains an argument with conclusion ψ^*.

Definition 13. *Let $\mathcal{C} = (K, \mathsf{P})$ be a probabilistic causal model. For the counterfactual statement $\phi \mathrel{|\!\!\sim}_{\mathcal{C}^*_{[v^*=x]}} \psi^*$, the probability that ψ would have been true, given ϕ and assuming $v{=}x$, is computed as*

$$P(\phi \mathrel{|\!\!\sim}_{\mathcal{C}^*_{[v^*=x]}} \psi^*) = \sum_{S \in \mathsf{S}_{[\psi^*=true]}} P_{\mathsf{AF}}(S).$$

The probabilistic argumentation-based twin network approach is equivalent to Pearl's standard three-step procedure.

Theorem 2. *Let $\mathcal{C} = (K, \mathsf{P})$ be a probabilistic causal model. Given a counterfactual statement $\phi \mathrel{|\!\!\sim}_{\mathcal{C}^*_{[v^*=x]}} \psi^*$, we have that $P(\phi \mathrel{|\!\!\sim}_{\mathcal{C}^*_{[v^*=x]}} \psi^*) = P_{\mathcal{C}}(\psi \mid \phi)$.*

5 Discussion

In this work, we extended our argumentation-based approach for defeasible causal and counterfactual reasoning from [1] to the probabilistic scenario. The intention of our approach is to bridge the gap from causal reasoning to formal argumentation. Our approach provides an argumentative representation of the causal mechanisms of the model in the context of a specific causal or counterfactual statement. In the literature, approaches for generating explanations for the (non-)acceptance of arguments in an argumentation framework have already been proposed [5]. The work [8] introduces a new kind of semantics called *related admissibility* which computes sets of arguments that are related to a specific argument. These sets form the basis of different kinds of explanations for the argument. Based on the same idea, they also introduce *dispute forests* that can be used to explain the non-acceptance of an argument. Furthermore, [3] introduce a general framework for explanations in formal and structured argumentation. They define different kinds of explanations, for example, an explanation for or against an argument as well as evidence that supports or is incompatible with an argument. This approach is especially interesting since they also consider the structured argumentation formalism ASPIC$^+$ [15], which is very similar to how we induce argumentation frameworks from causal models in our approach.

There also exist other argumentation-based approaches in the literature that highlight the interest in explaining causal reasoning. For instance, the work [18] is concerned with Bayesian networks and introduces the notion of a support graph that makes d-separation explicit, which eliminates circular causal structures and helps to explain interdependent causes.

In a recent work [16], Rago et al. introduce an approach for generating bipolar argumentation frameworks from causal models in the sense of Pearl. They create so called explanation moulds, that reinterpret desirable properties of semantics of argumentation frameworks. In their approach, they interpret causal atoms directly as arguments and causes contribute positively or negatively towards arguments via attack and support relations, respectively.

6 Limitations

In the following we discuss the limitations of the approach introduced in this work. First, our approach is built on classical propositional logic. That means, while being relatively easy to understand, the expressiveness is limited when compared to other higher-order logics.

Our approach is only focused on the actual reasoning with a causal model. That means we consider the underlying causal model to be given and crafted by experts and we assume that the given relations between the variables are indeed causal and not merely correlations.

Furthermore, the computational complexity of this approach to causal reasoning is quite high. Our approach relies on deciding whether some of the arguments are skeptically accepted in the induced argumentation framework. This problem is naturally difficult and in the case of the stable semantics that we use it has been shown to be NP-complete [7]. In addition to that, when considering probabilistic causal reasoning we have to potentially consider exponentially many framework states (wrt. the set of background variables) which increases the complexity of the approach significantly.

Finally, it should also be noted that our approach is to be understood as a groundwork for making causal reasoning explainable. Meaning the induced (probabilistic) argumentation framework can be the basis for crafting human understandable explanations. How exactly these explanations should look like, is left for future work and some interesting approaches for that matter have already been highlighted in Sect. 5. Especially in the case of probabilistic causal reasoning this is even more difficult since the probabilistic aspect has to be somehow incorporated into the explanations.

7 Conclusion

We extended our approach for argumentation-based causal reasoning from [1] to deal with probabilistic causal models. For that, we model probabilistic causal reasoning in a probabilistic argumentation framework and compute the probability that the statement is true by reasoning in the framework states. Furthermore, we showed that our approach can also be used for reasoning with counterfactuals, by adapting Pearl's twin network method. Besides computing the probability, the generated probabilistic argumentation framework can be used as the basis for creating explanations of the underlying causal mechanisms of the model in the context of the statement, since it provides both arguments supporting the prediction as well as arguments that refute the prediction.

Future work includes determining structural properties of the generated (probabilistic) AFs and looking into concrete application scenarios to investigate the capabilities of our approach.

Acknowledgement. The research reported here was partially supported by the Deutsche Forschungsgemeinschaft (grant 375588274).

References

1. Bengel, L., Blümel, L., Rienstra, T., Thimm, M.: Argumentation-based causal and counterfactual reasoning. In: 1st International Workshop on Argumentation for eXplainable AI, Cardiff. CEUR Workshop Proceedings, vol. 3209 (2022)
2. Bochman, A., Lifschitz, V.: Pearl's causality in a logical setting. In: Bonet, B., Koenig, S. (eds.) Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015, pp. 1446–1452. AAAI Press (2015)
3. Borg, A., Bex, F.: A basic framework for explanations in argumentation. IEEE Intell. Syst. **36**(2), 25–35 (2021)
4. Cayrol, C.: On the relation between argumentation and non-monotonic coherence-based entailment. In: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI 1995), pp. 1443–1448 (1995)
5. Čyras, K., Rago, A., Albini, E., Baroni, P., Toni, F.: Argumentative XAI: a survey. arXiv preprint arXiv:2105.11266 (2021)
6. Dung, P.M.: On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. Artif. Intell. **77**(2), 321–358 (1995)
7. Dunne, P.E., Wooldridge, M.: Complexity of abstract argumentation. Argumentation in Artificial Intelligence, pp. 85–104 (2009)
8. Fan, X., Toni, F.: On computing explanations in argumentation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 29 (2015)

9. Hunter, A.: A probabilistic approach to modelling uncertain logical arguments. Int. J. Approx. Reason. **54**(1), 47–81 (2013)
10. Hunter, A., Polberg, S., Potyka, N., Rienstra, T., Thimm, M.: Probabilistic argumentation: a survey. Handb. Formal Argument. **2**, 397–441 (2021)
11. Madumal, P., Miller, T., Sonenberg, L., Vetere, F.: Explainable reinforcement learning through a causal lens. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 2493–2500 (2020)
12. Manor, N.R.R., Rescher, N.: On inference from inconsistent premises. Theor. Decis. **1**, 179–219 (1970)
13. Minh, D., Wang, H.X., Li, Y.F., Nguyen, T.N.: Explainable artificial intelligence: a comprehensive review. In: Artificial Intelligence Review, pp. 1–66 (2022)
14. Pearl, J.: Causality: Models, Reasoning and Inference, vol. 29. Cambridge University Press (2000)
15. Prakken, H.: An abstract framework for argumentation with structured arguments. Argument Comput. **1**(2), 93–124 (2010)
16. Rago, A., Russo, F., Albini, E., Baroni, P., Toni, F.: Forging argumentative explanations from causal models. In: Proceedings of the 5th Workshop on Advances in Argumentation in Artificial Intelligence 2021). CEUR Workshop Proceedings, vol. 3086. CEUR-WS.org (2021)
17. Rajkomar, A., et al.: Scalable and accurate deep learning with electronic health records. NPJ Digit. Med. **1** (2018)
18. Timmer, S.T., Meyer, J.-J.C., Prakken, H., Renooij, S., Verheij, B.: Explaining Bayesian networks using argumentation. In: Destercke, S., Denoeux, T. (eds.) ECSQARU 2015. LNCS (LNAI), vol. 9161, pp. 83–92. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-20807-7_8

From Networks to Narratives: Bayes Nets and the Problems of Argumentation

Anita Keshmirian[1,2,6], Rafael Fuchs[2,3](✉), Yuan Cao[1,4], Stephan Hartmann[2], and Ulrike Hahn[2,5]

[1] Fraunhofer Institute for Cognitive Systems (IKS), Munich, Germany
anita.keshmirian@gmail.com
[2] Munich Center for Mathematical Philosophy (MCMP) - LMU, Munich, Germany
Rafael.Fuchs@campus.lmu.de
[3] Graduate School of Systemic Neurosciences (GSN) - LMU, Munich, Germany
[4] Technical University of Munich, Munich, Germany
[5] Birkbeck University of London, London, UK
[6] Forward College, Berlin, Germany

Abstract. Bayesian Belief Networks (BBNs) are gaining traction in practical fields such as law and medicine. Given this growing relevance, it is imperative to make Bayesian methodologies accessible to professionals in these fields, many of whom might lack formal training in probability calculus. Argumentation offers a promising avenue to achieve this. It serves a dual purpose: (i) generating an explanation of the important reasoning steps that occur in Bayesian inference and (ii) exploring the structure of complex problems, which can help to elicit a BBN representation. Since Bayesian probabilistic inference also provides clear normative criteria for argument quality, there is a tight conceptual connection between the argumentative structure of a problem and its representation as a BBN. The primary challenge is representing the argumentative structure that renders BBN inference transparent to non-experts. Here, we examine algorithmic approaches to extract argument structures from BBNs. We critically review three algorithms - each distinguished by its unique methodology in extracting and evaluating arguments. We show why these algorithms still fall short when it comes to elucidating intricate features of BBNs, such as "explaining away" [44] or other complex interactions between variables. We conclude by diagnosing the core issue and offering a forward-looking suggestion for enhancing representation in future endeavors.

Keywords: Bayesian Network · Argument Quality · Explainable AI

Supported by Deutsche Forschungsgemeinschaft (DFG) - Project number 455912038 and the Bavarian Ministry for Economic Affairs, Regional Development, and Energy as part of a project to support the thematic development of the Fraunhofer Institute for Cognitive Systems IKS.

A. Keshmirian and R. Fuchs—Equally contributed to this work.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 237–252, 2024.
https://doi.org/10.1007/978-3-031-63536-6_14

1 Introduction

At the heart of scientific discovery and societal progress lies a fundamental concept: argumentation. It plays a pivotal role not only in the realm of science for explanation, justification, and the discovery of scientific laws but also serves as the backbone of effective communication and decision-making across disciplines, making it indispensable for a well-functioning democracy.

What constitutes a good argument? This enduring question has fascinated scholars for centuries, yielding a diverse array of studies and theories (e.g., [7, 15, 23, 34, 41, 43] , with a comprehensive review available in [13].

In recent decades, Bayesian argumentation has emerged as a significant methodological breakthrough, offering explicit, normatively grounded criteria for evaluating arguments' quality and inferential soundness [14]. Its foundational principle in practical and scientific reasoning [1,2] and its focus on minimizing the retention of incorrect beliefs [32] mark it as a notable advance in argumentation theory.

A particularly promising tool within the Bayesian framework is Bayesian Belief Networks (BBNs). These probabilistic models visualize complex relationships between variables and their dependencies while enabling sophisticated inference and learning capabilities about uncertain outcomes [21,31]. This investigation zeroes in on three innovative algorithmic strategies-Sevilla [38], Timmer et al. [40], and Keppens [19] - for extracting and scrutinizing arguments within BBNs. By dissecting these methods, we aim to illuminate their strengths and limitations, advocating for a paradigm shift towards more effective argumentation methodologies. Our critical analysis highlights the challenges inherent in managing the 'Explaining Away' effect and the nuanced treatment of 'Soft Evidence', underscoring the need for a sophisticated understanding of evidential relations within BBNs.

The paper is structured as follows: We begin by exploring the fundamentals of BBNs, spotlighting their role in modeling argumentation through the lens of The Spider network [4,33], an example network designed to test decision-making in uncertain contexts using BBNs [4,33].

We then critically compare three algorithmic approaches to argument extraction, exposing ongoing challenges and the need for clearer methodologies to tackle the nuanced dynamics of argument strength and validity in interconnected systems. Our analysis emphasizes the crucial role of transparency and accessibility in these methodologies, aiming to demystify the complexities for both scholars and practitioners.

2 The Bayesian Approach to Argumentation

This section gives a brief overview of the Bayesian approach to argumentation. Section 2.1 outlines the motivation and generality of the Bayesian approach, highlighting important advances in developing formal tools. Section 2.2 introduces Bayesian Belief Networks (BBNs) as a powerful graphical tool within the

Bayesian framework, which can significantly simplify computations and help visualize relations of conditional (in)dependence. Finally, in Sect. 2.3, we present important challenges when it comes to explaining the reasoning with BBNs to non-experts. Here, we primarily focus on the Explaining Away effect and why it is difficult to grasp intuitively. The effect can be further modified by the presence of Soft Evidence, which raises the probability of an observation without becoming fully certain.

2.1 The Bayesian Framework

Revisiting our primary query, we ask: what defines a compelling argument? The existence of argumentative *fallacies* shows that subjective (psychological) persuasiveness and objective argument quality can come apart. Fallacies were studied extensively in arguments that psychologically seem persuasive, but as we examine them, we realize that they *should not* convince us [15,45].

However, interestingly, arguments that share the same form with other fallacious arguments can still be good arguments in a different context [11]. For instance, compare the two arguments: "We haven't discovered any extraterrestrial life so far. Therefore, there is no extraterrestrial life in our universe." versus "After several checks, we couldn't discover any technical problems with this engine. Therefore, the engine works." Both arguments have an analogous form, known as *argument from ignorance*: given the absence of evidence to the contrary, we accept the hypothesis. However, while the former argument seems quite strong (at least to the extent that the premise can grant the conclusion), the latter is entirely reasonable, and in fact, we (have to) rely on this kind of reasoning all the time. Arguments from ignorance—one example of many informal argument schemes—have been discussed in the literature [12,30].

This underscores that argument quality in realistic scenarios isn't solely based on syntactic form. Approaches to argumentation focusing only on syntactic form or deductive validity overlook critical elements of real-world argumentation, such as belief dynamics, graded uncertainty about propositions, and the interaction of relevant factors. Arguments in real-world scenarios are shaped by uncertain evidence, the audience's prior beliefs, and information source reliability [12,14]. Thus, a purely deductive approach may not sufficiently address these real-world complexities.

This brings us to the necessity of a more adaptive approach. A probabilistic approach, capable of addressing uncertainty and still containing deductive validity as a limiting case, emerges as promising. This approach, while retaining the foundations of logic, extends its capabilities. It enables a more in-depth evaluation of informal arguments, like in our example above, while aligning more with the multifaceted nature of real-world discourse. Furthermore, a probabilistic approach can help to unify the large plethora of argument schemes [42] that have been identified in studies of informal logic [10], showing how Bayesian reasoning can be used to explain how, when, and why diverse argument schemes actually work. There are also generalizations of standard Bayesian updating, such as Jeffrey conditionalization [18], which allows for updating credences based on uncertain evidence[1], and *distance-minimization* approaches [8,9], which also enable updating on more complex, non-propositional constraints.

In a nutshell, the strength of an argument is a question of *relevance*, expressed in terms of probability- or belief change. Argument quality is assessed by considering how one's belief in a target proposition would change upon learning the particular premise- an idea aptly encapsulated in the slogan of "argumentation as learning" [8].

Finally, knowledge bases that determine dependencies between propositions or variables of interest can be graphically represented as BBNs, which are directed acyclic graphs equipped with a probability distribution (more on this right below). BBNs are not just widely used in scientific contexts (e.g., [28]); they have found increasingly widespread adoption in software systems in domains as diverse as law, medicine, risk analysis, engineering, or strategic decision-making (e.g., [3,24]). BBNs are popular because they can provide relatively simple, compact representations of complex problems. Crucially, BBNs can not only be learned from data under certain assumptions [16,37] but are formulated in terms of the variables that figure in the discourse and theories of these domains. This contrasts with low-level, data-driven 'black box' systems and makes BBNs key candidates for the development of explainable AI systems (XAI) (regarding the difficulties of developing explanations for 'black box' systems see, e.g., [36] and for discussion of the broader role of argumentation for XAI see, e.g., [39]). Now, let us look more closely at BBNs' features and non-experts' difficulties understanding their dynamics.

2.2 Bayesian Belief Networks (BBNs)

BBNs provide a graphically compact and computationally powerful representation of complex problem domains (for an example, see Fig. 1). BBNs are graphs with multiple nodes (variables), directed edges (relationship between variables), and no cycles. In these Directed Acyclic Graphs (DAGs), directed edges connect parent nodes to their downstream child nodes with a conditional probability table for each Parent-Child relationship. This allows a compact, computationally efficient encoding of the joint probability distribution as a product of conditional probabilities [22,25]. This specific feature is called the Markov property. It states that each node is conditionally independent of its non-descendants (i.e., ancestors or unrelated nodes) in the network, given its parents.

BBNs were shown to mitigate some frequent biases in reasoning under uncertainty [4,29]. Having an intuitively understandable tool is highly beneficial in the domains mentioned above because experts (e.g., in law) are not necessarily experts in probability calculus, and therefore, the quality of decisions can be improved by providing adequate and understandable explanations of correct probabilistic inferences.

Let us consider the following example, known as the 'Spider' scenario [33]:

[1] Without raising the probability of the observed variable to total certainty.

Imagine you're an intelligence analyst on the trail of a dangerous foreign spy known as 'the Spider.' Initial evidence suggests that the Spider might be hiding in a facility located in a neutral country, represented by the binary variable Sp. Your mission is to collect more information to determine if your team should infiltrate the facility to apprehend the Spider. You receive reports from agents Emerson and Quinn indicating the Spider's presence in the facility (binary variables E, Q), known for their reliability (low false-positive and false-negative rates). However, you soon encounter telephone logs that suggest Emerson and Quinn could collaborate with the Spider (binary variable L), though there's a chance these logs are forgeries created by the Spider's allies to mislead. To effectively navigate this situation, you must synthesize all these pieces of evidence, particularly considering the potential disinformation $L = l$ that could affect the credibility of positive reports from Emerson and Quinn, $E = e$, and $Q = q$.

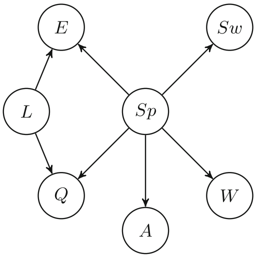


Fig. 1. The Spider Network

While a BBN's graphical representation makes the overall independence structure and potential causal relations directly visible, other, more indirect reasoning features still challenge humans. In the following, we present two exemplary features of BBNs that are hard to understand: uncertain (soft) evidence, explaining away, and synergistic interaction effects.

2.3 Explaining BBNs: Important Challenges

Now, we present two factors that complicate the explanation of BBNs: Explaining Away and Soft Evidence. We start with the latter and then show how it affects Explaining Away (our main focus), which we introduce next.

Soft Evidence. refers to cases in which an event is not learned or observed with *certainty* (i.e., probability 1), but its probability only *increases*. This can also happen if a leaf node is observed, but we are interested in its effect on further upstream nodes. Hence, we must calculate the effects of intermediate nodes, whose probability is raised by observing the respective leaf nodes, but it still remains below certainty. Within the Bayesian framework, we can accommodate soft evidence as a case of Jeffrey conditionalization [18], but for the untrained user, it may be difficult to track how the probability flow propagates without further visualization or explanation.

Notably, soft evidence can also reverse the explaining-away effect, which we explain next.

Explaining Away. is a particular (and potentially tricky) effect that occurs in collider networks $X \rightarrow Z \leftarrow Y$. The collider network represents a structure where an effect has several possible, unconditionally independent causes[2]. Suppose the effect Z is observed, and the probability of one of the possible causes increases (say X). This can lead to a *decrease* in the probability of the alternative explanation – it is "explained away" by the presence of the first cause. Pearl [31] provides an intuitive example of this effect: Suppose your car's failed (variable Z), and the possible causes are either a dead battery (X) or a blocked fuel pump (Y). If you learn that the battery is dead, this is a sufficient explanation of your observation regarding Z – if the fuel pump was blocked as well, this would be a very unlucky coincidence, and hence, you may think in this kind of situation that X "explains away" Y. On the other hand, there are cases with a similar structure where we might be intuitively inclined to think that learning about one cause doesn't give us any further information regarding the other. So, what is going on here?

The general answer is that we can observe this effect in collider networks with binary variables X, Y, Z (with values $x, \neg x$, i.e., the negation), equipped with a probability distribution that satisfies the following inequality:

$$P(z|x, y) \cdot P(z|\neg x, \neg y) < P(z|\neg x, y) \cdot P(z|x, \neg y) \tag{1}$$

If this holds, then observing Z makes X and Y dependent in the following sense:

- If the probabilities of $X = x$ and $Y = y$ increase due to $Z = z$, then $P(x|z) > P(x)$ and $P(y|z) > P(y)$.
- If either $Y = y$ or $X = x$ is observed in addition to $Z = z$, then $P(x|z, y) \leq P(x|z)$ or (respectively) $P(y|z, x) \leq P(y|z)$.

The DAG by itself does not indicate whether this holds, which means that the user needs to understand the probabilistic relations or the graphical representation (or AI-generated verbal explanation) needs to be adequately extended to include this information understandably. For humans, explaining away often seems to be challenging to grasp. Several empirical studies have indicated subjects' tendency to under-update or contradict the prescription of explaining-away-reasoning (e.g., [35]). Thus, it is an important desideratum for explanatory algorithms to make this relation apparent to the user and, ideally, provide an explanation that fits the context of the application.

[2] Here, we are explicitly referring to uncoupled colliders, where there is no link between X and Y.

Finally, as mentioned above, soft evidence can *reverse* 'explaining away' as follows. In a collider $X \rightarrow Z \leftarrow Y$, if there is soft evidence for Z, i.e., $P(Z = z)$ increases, then X can confirm Y. This makes the intuitive understanding even more difficult because the BBN does not depict the probability flow. Hence, we need the relevant background knowledge to draw the correct inference from observing a BBN and facts about changing probabilities. An explainable algorithm thus has to provide the relevant background information and make it salient in the context of the given application scenario.

3 Algorithmic Approaches to Bayesian Argumentation

This section reviews the relation between argument diagrams (ADs) and BBNs and three extant approaches to AD extraction from BBNs. Specifically, in Sect. 3.1, we analyze general conceptual questions about the relation between ADs and BBNs, as well as general desiderata for explanatory or auxiliary ADs. In Sect. 3.2, we present three extant approaches to argument extraction: the factor graphs by Sevilla [38], the support construction by Timmer [40], and the argument-diagram-extraction by Keppens [19]. In Sect. 3.3, these algorithms are evaluated using the Spider example, which we introduced previously (Fig. 1). We find that these algorithms illuminate the connection between argument diagrams and BBNs, but ultimately, the main challenges identified in Sect. 2.3 remain unresolved. Further work combining different approaches and extended psychological research will be needed to develop a more comprehensive and theoretically well-founded approach to explanatory reasoning with BBNs.

3.1 The Relation Between Argument Diagrams and Bayesian Networks

An argumentative problem-solving approach often helps to increase the understanding of complex problems. This also comes out in the social procedure of the BARD project [29], designed for the *elicitation* of BBN representations via group deliberation. We have already seen that BBNs (as mathematical objects) contain features that are intuitively challenging for laypeople, but when problems are posed within a context of practical argumentation, correct reasoning and intuitive understanding can be improved [17]. The interaction between inference in BBNs and argument diagramming techniques becomes interesting at this point. Experts in practical domains, e.g., in law, tend to understand argument diagrams better than causal models or BBNs [19,40]. At the same time, inference with BBNs provides a normative standard for correct reasoning under uncertainty. Therefore, BBNs and more informal argument diagramming techniques can exhibit synergies that benefit the general project of widening access to Bayesian reasoning resources.

Generally, the information exchange between BBNs and Argument diagrams (ADs) can go in both directions:

1. **Elicitation** (from ADs to BBNs): an argumentative exchange about a target domain or problem (represented as an AD) is mapped to a BBN. This requires an unambiguous mapping from input ADs and additional technical constraints (accounting for contextual and pragmatic factors in conversation) to BBNs.
2. **Explanation** (from BBNs to ADs): probabilistic inference in a BBN is transferred to an AD, which can then also serve as the basis for verbal explanations and be supplemented with quantitative impact measures (how much each premise or piece of evidence impacts the set of target variables or conclusions).

The literature on algorithmic argument generation and explainable AI has generated some approaches to algorithmic argument extraction from BBNs. In the following, we review frameworks by Sevilla 2021 [38], Timmer [40], and Keppens [19].

3.2 Introducing Three Extant Algorithms

Sevilla (2021). This algorithm, developed by Jaime Sevilla [38], finds an approach to select a list of relevant and independent arguments from a BBN, given evidence nodes and a target node. The strength of each argument is computed by the logarithmic odds ratio, calculated after implementing the approximate message-passing algorithm to ascertain the relatively important arguments. The algorithm generates a *factor graph* [22] from a BBN with the nodes for all variables in the model and the factors representing the conditional probability tables. The nodes connected to the factors are part of the conditional probabilities. To prepare the message passing calculation, each observation node is initialized to a lopsided factor (only the known state with probability one and others with probability 0). In contrast, the remaining nodes are initialized to constant factors (under uniform distribution). After obtaining the factor graph, the message-passing algorithm could estimate all message flows. Effects and strength of argument: An argument is indicated as a directed acyclic graph over a factor graph consisting of nodes and factors from observation to the target. The effect of each inference step in an argument defines how a preceding node affects the inferior node. The factor is multiplied by all premises as the message-passing algorithm and then divided by the factor itself to distinguish between the information obtained from the updates Δ and the information inherently embedded within the conditional probability table ϕ.

The effect of a complete argument is calculated by recursively utilizing the Step Effect. The effects of all the parents of the factor are multiplied together to inherit the effect of an argument. The strength of an argument is introduced to compare the importance of all the arguments. It is the logarithmic odds of the argument that support the outcome. This measure of strength is a real-valued quantity, where its sign indicates whether the argument supports or opposes the outcome, and its magnitude quantifies the strength of the argument.

Argument independence: to decide whether simple arguments should be combined into one complex argument indicates to determine whether they are independent. More ordinarily, a list of arguments is independent if the effect of the union of arguments is equal to the product of the effects of simple arguments. To adjust the theory to reality, a list of arguments is approximately independent if the distance of effects is within a certain threshold.

The final output presented to the user is a text generated from basic blocks which take premises (evidence nodes) and a query node as input and give an evaluation of how much the given set of premises supports the probandum, with the logarithmic odds as a measure of argument strength (supplemented by a qualifier tag, such as 'weak inference' or 'strong inference').

Timmer (2017) focuses on the construction of *support graphs* from BBNs. Support graphs are trees with a given query node as their root, and the descendants on each branching layer consist of all the variables that directly affect their parent. This tree preserves the conditional independence structure, which entails that all Markov-equivalent DAGs map onto the same support graph. This approach promises that a tree, in which the conclusion is at the top, and the supporting evidence (premises) are on the layers below, is easier to interpret than a BBN, in which we sometimes have to reason 'backward' (e.g., from effects to possible causes, as in the explaining away effect). Due to the close connection between support graphs and BBNs, Timmer's construction promises to be a good candidate for *elicitation*, i.e., a stepwise translation of (informal) argument diagrams into BBNs.[3]

The variables are mapped from the BBN to the new structure to construct a support graph with the query node (conclusion) as a root. In doing so, the same variables occur multiple times on the tree. Therefore, to avoid the inclusion of false independencies and (in the extreme case) circular reasoning, a set of *forbidden nodes* is defined, whose purpose is to exclude these problematic instances. The set of descendants for each given node in the tree is then defined as the Markov blanket of the corresponding node in the BBN minus the set of forbidden variables. The algorithm terminates when no further nodes are added. Formally, the set $\mathcal{F}(V_i)$ of forbidden nodes for variable V_i is defined as follows.

- $\mathcal{F}(V_i) = \{V_i\}$, if V_i is the query node (root of the SG)
- Otherwise, if V_j is a parent of V_i in the SG:
  - $\mathcal{F}(V_i) = \mathcal{F}(V_j) \cup \{V_i\}$, if V_i is a parent of V_j in the BBN
  - $\mathcal{F}(V_i) = \mathcal{F}(V_j) \cup \{V_i\} \cup Par(V_i)$, if V_i is a child of V_j in the BBN
  - $\mathcal{F}(V_i) = \mathcal{F}(V_j) \cup \{V_i\} \cup \mathbf{C}_{i,j}$, otherwise, where $\mathbf{C}_{i,j}$ are the *common children* of V_i, V_j in the BBN.

[3] In the interest of space, we cannot cover this aspect here, but we note that it is an interesting direction for future research.

Keppens (2013) considers arguments as consisting of observed variables. The input of the algorithm is a BBN, together with a query node (probandum), denoted as $H = h$, and a set of observations $\mathbf{O} = \{O_1 = o_1, ..., O_n = o_n\}$ (the evidence). Given this initial set, the algorithm finds all nodes on a path between (i) one of the variables corresponding to observations and (ii) the probandum in the BBN. For these intermediate nodes, the algorithm calculates the most probable value (defined as $\arg\max_{v \in Im(V)} P(v|\mathbf{O}, h)$), given the values of probandum and observations. Finally, the set of edges in Keppens' AD corresponds precisely to the set of edges of the BBN, but the edges in the baseline AD are inverted. This is due to the intended application to forensic reasoning, where we must standardly reason backward, from evidence (observations) to the most likely explanatory hypothesis (probandum).

Furthermore, Keppens' algorithm has additional features that potentially make his AD formalism more expressive than Timmer's. In particular, we point out the distinction between *convergent* and *linked* arguments. Convergent and linked arguments are arguments that share the same conclusion. Convergent arguments have independent sets of premises, while in linked arguments, there is dependence among the premises. Keppens' proposed criterion to distinguish between convergent and linked arguments is whether the variables corresponding to premises are d-separated by the conclusion in the BBN (if yes, the arguments are convergent; otherwise, they are linked). In the AD, these can be represented via a single hyper-edge[4] that connects a set of linked (dependent) arguments to a single claim or separate edges pointing from individual convergent (independent) arguments to one conclusion.

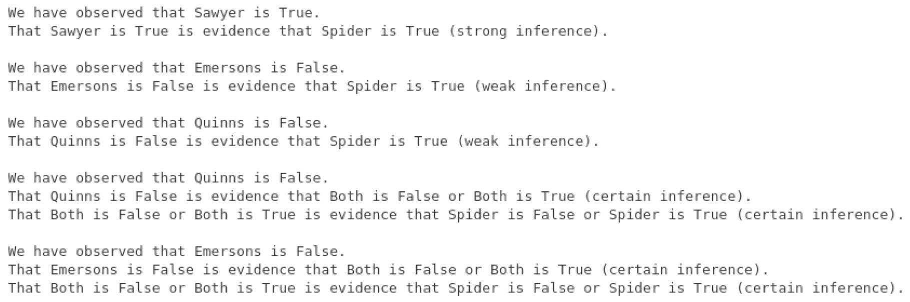
The final step is decorating the inference links in the resulting AD with labels that indicate *probative force*. These are verbal descriptions ('strong', 'weak', 'certain' etc.) based on intervals of likelihood ratios.

3.3 Evaluating the Algorithms: Example Networks

Let us examine how well these algorithms connect to BBNs, focusing on *explanation* (noting that exploring the potential of support graphs for elicitation remains for future work). We consider an example case, "The Spider" [4,33], to evaluate how well the algorithms fare regarding explanation. This example stands out for its prior demonstrations of challenging human and artificial agents with its nuanced scenario, often revealing areas of sub-optimal reasoning.

Sevilla: Sevilla's algorithm yields the output shown in Fig. 2. The algorithm can quantify individual support relations and provide an overall evaluation (not shown here). Still, it is not designed to show precisely how arguments interact. In particular, the dynamics related to explaining away do not appear, and the simple list of 'strength values' might look rather confusing to an untrained user. Furthermore, the algorithm produces artifacts that output irrelevant conclusions (such as the 'certain inference' in the last two paragraphs).

[4] a hyper-edge connects a set of nodes to another node or set of nodes.



```
We have observed that Sawyer is True.
That Sawyer is True is evidence that Spider is True (strong inference).

We have observed that Emersons is False.
That Emersons is False is evidence that Spider is True (weak inference).

We have observed that Quinns is False.
That Quinns is False is evidence that Spider is True (weak inference).

We have observed that Quinns is False.
That Quinns is False is evidence that Both is False or Both is True (certain inference).
That Both is False or Both is True is evidence that Spider is False or Spider is True (certain inference).

We have observed that Emersons is False.
That Emersons is False is evidence that Both is False or Both is True (certain inference).
That Both is False or Both is True is evidence that Spider is False or Spider is True (certain inference).
```

Fig. 2. Argument obtained from adding Emerson's and Quinn's reports as premises for the conclusion that the Spider is in the facility.

Timmer: Timmer's algorithm yields the output shown in Fig. 3. These support graphs are not particularly informative regarding the interaction between the evidence pieces. All of them are in the Markov blanket of Sp. Therefore, all are directly relevant to the conclusion—but from the graph alone, we don't know how. In particular, the explaining away effect between Sp and L that is triggered by increasing $P(Sp = true)$ via $W = true$ is not visible in the graph. Similarly, the support graph doesn't show how the explaining away relation changes under soft evidence: recall that soft evidence can reverse the explaining away effect. However, neither of these effects is visible in the support graph since its structure is always the same. Since the support graph is limited in this way, also a verbal explanation that is based *only* on information provided by the support graph (i.e., 'translating' the support graph into a textual explanation via some fixed scheme) cannot make these effects visible either—precisely because the relevant information is missing. Hence, the support graph does not add much explanatory power to the BBNs, except for clarifying which variables directly affect the target node.

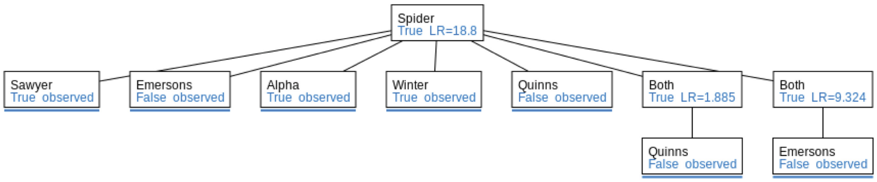


Fig. 3. Full support graph with observed reports by Winter and Alpha in the Spider Network.

Keppens: The core structure of Keppens' argument diagram in this scenario looks as follows:

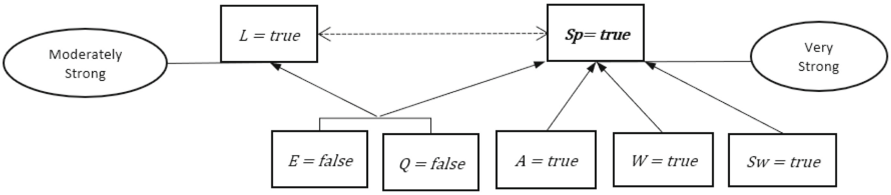


Fig. 4. Argument Diagram extracted with Keppens' algorithm.

In this argument diagram, the differentiation looks better than in Timmer's case because $L = true$ and $Sp = true$ are depicted as alternative explanations of evidence pieces $E = false$ and $Q = false$ (represented as a bidirectional edge between nodes L and Sp). $E = false$ and $Q = false$ are linked arguments. Therefore, they are connected via a hyperedge to both alternative conclusions.

The limitations in Keppens' case primarily relate to the aggregation and the assignment of labels of probative force. When assessing how a specific piece of evidence influences the probability of the target node and when comparing the situation before and after observing that evidence, it is crucial to understand how the probative force of the total (aggregated) argument shifts concerning the target node. For example, if we start with the reports from Emerson and Quinn, their joint probative force for Spider being in the facility may be "strong." So, in the next step, we need to check how this assignment changes when we add Winter's (and other witnesses') reports. In the best case, the label changes (e.g., from "strong" to "very strong"), which, if not numerically precise, gives the user at least a qualitative understanding of how the respective variables in the BBN interact. However, the probability shift happens within the interval in the worst case. Thus, the final label may still say "strong" even though there was a shift from the lower bound of "strong" to the higher bound (i.e., *almost* "very strong"). Thus, it is still a non-trivial question of how this can be represented in an argument diagram without numerical values. A complete solution must include numerical values and verbal interpretations adequate for context.

Another limitation concerns the representation of linked arguments. In a collider $X \rightarrow Z \leftarrow Y$, an argument based on X and Z to the conclusion Y is linked (the premises are not d-separated by the conclusion Y), but their *roles* are fundamentally different. While we can remove premise X, and still argue for/against some value of Y only with Z, the reverse is impossible since X and Y are unconditionally independent. Thus, the classification of linked arguments must be refined for more advanced applications and faithful representation of BN relations in an explanatory argument diagram.

These results indicate that simple, static displays of AD still have limited explanatory power regarding dynamic interactions between evidence variables since probabilistic information flows back and forth in the BBN, as exemplified in the previous section. However, as illustrated by Keppens' none of these problems seem unsolvable. Thus, we are optimistic that future work (taking Keppens' algorithm as a starting point and refining it further) can solve at least a good portion of the persisting challenges. As Sevilla used, additional text generation seems indispensable to generate more complete explanations. However, the outputs produced by Sevilla's algorithm show that significant challenges still exist to come closer to a comprehensive solution.

Alternative methods for explaining Bayesian networks have also been proposed, for example, interactive graphical explanations that use color codings and node- and edge sizes to indicate interactions between variables (see, e.g., [20]). Suppose the graphical display of argument structures extracted from Bayesian networks can be useful. In that case, such an interactive approach might be better—but it is unclear whether this would have any value over just using a more colorful and interactive version of the Bayes net itself. However, the generation of *text* from argumentative seems promising because it goes beyond the graphical modality and adds a new dimension to help the user understand from a different perspective. Text can combine a linear path of reasoning with changing interactions, feedback, and back-and-forth that occurs as more information is introduced. This can be done on a high level (thus being able to handle large networks without getting lost) or in a more detailed way. So far, text-based algorithms are still undeveloped (no new recent approaches were presented besides Sevilla's), and thus, we believe that pushing this line of research will be fruitful in the future.

4 Limitation

In this study, we selectively examined three diverse methodologies, acknowledging the existence of additional approaches that could further enrich our analysis. This deliberate choice allowed us to showcase a range of distinctly different methods, setting a foundation for comprehensively exploring this field. While our investigation focused on a carefully chosen example to illuminate specific challenges, it opens the door to examining other networks, particularly larger ones, in future studies. Our critical review of current algorithms lays the groundwork for future research to build upon, presenting an exciting opportunity to develop innovative solutions to the challenges we have highlighted.

This paper represents a significant stride towards refining our approach to Bayesian Belief Networks (BBNs) and their applications. At its heart, the comparison of three distinct approaches serves not only to highlight the current state of the art but also to spark a deeper inquiry into methodological enhancements. Notably, while our analysis suggests that translating a BBN into an argument structure could potentially deepen our understanding of probabilistic inferences, it also points to an intriguing area for future empirical investigation. This prospect underscores the forward-looking nature of our work, inviting further research to validate these claims and continue advancing the field in novel and meaningful directions. In non-written argumentation paradigms mentioned in the paper, we need to note that the cognitive capacity of human end users is inherently limited, which presents a notable challenge to the scalability of graphical approaches, as their effectiveness might diminish when applied to larger and more complex models. Future research should consider a fundamental shift in approach, aligning more closely with the intricacies of argumentation and Bayesian reasoning. This would ensure that the powerful potential of BBNs can be fully harnessed in diverse real-world applications.

5 Conclusion

In our exploration of algorithmic methods to extract and evaluate arguments from Bayesian Belief Networks (BBNs), we identified persistent challenges, particularly concerning the intricate features of BBNs. The difficulties in capturing nuances like interdependence underscore the complexities inherent to these probabilistic models. While these algorithms provide valuable insights and bring us closer to making Bayesian methodologies more comprehensible, our analysis indicates that more innovative approaches are needed. A holistic understanding of the argumentative structure is crucial for transparent BBN inference, especially for those without expert knowledge in probability calculus. As BBNs gain prominence in decision-making across disciplines, refining these algorithms is not just an academic endeavor but a practical necessity.

Acknowledgement. We thank the Deutsche Forschungsgemeinschaft (DFG) and the Fraunhofer Institute for Cognitive Systems IKS for supporting our research. We thank Dr. Narges Ahmidi and Alireza Zamanian for their insightful suggestions and valuable feedback on this paper.

References

1. Corner, A., Hahn, U.: Evaluating science arguments: evidence, uncertainty, and argument strength. J. Exp. Psychol. Appl. **15**(3), 199 (2009)
2. Corner, A., Hahn, U.: Normative theories of argumentation: are some norms better than others? Synthese **190**(16), 3579–3610 (2013)
3. Coutts, A.: Balancing the validity and viability of Bayesian belief networks for the study of national strategic decisions. In: 22nd National Conference of the Australian Operations Research Society. ASOR, Adelaide (2013)
4. Cruz, N., et al.: Widening access to Bayesian problem-solving. Front. Psychol. **11**, 660 (2020)
5. Dalkey, N., Helmer, O.: An experimental application of the DELPHI method to the use of experts. Manage. Sci. **9**, 458–467 (1963)
6. Dewitt, S., Lagnado, D., Fenton, N.: Updating prior beliefs based on ambiguous evidence. In: Liefgreen et al., pp. 2047–2052 (2018)
7. van Eeemeren, F.H., Grootendorst, R.: A systematic theory of argumentation. In: The Pragma-Dialectical Approach. Cambridge University Press, Cambridge (2004)
8. Eva, B., Hartmann, S.: Bayesian argumentation and the value of logical validity. Psychol. Rev. **125**(5), 806–821 (2018)
9. Eva, B., Hartmann, S., Rad, S.R.: Learning from conditionals. Mind **129**(514), 461–508 (2020)

10. Hahn, U., Hornikx, J.: A normative framework for argument quality: argumentation schemes with a Bayesian foundation. Synthese **193**, 1833–1873 (2016)
11. Hahn, U., Oaksford, M.: A Bayesian approach to informal argument fallacies. Synthese **152**, 207–236 (2006)
12. Hahn, U., Oaksford, M.: The rationality of informal argumentation: a Bayesian approach to reasoning fallacies. Psychol. Rev. **114**, 704–732 (2007)
13. Hahn, U., Oaksford, M.: Rational argument. In: Morrison and Holyoak (eds.) Oxford Handbook of Thinking and Reasoning, pp. 277–298. Oxford University Press, Oxford (2012)
14. Hahn, U.: Argument quality in real-world argumentation. Trends Cogn. Sci. **24**, 363–374 (2020)
15. Hamblin, C.L.: Fallacies. Methuen, London (1970)
16. Heckerman, D.: A tutorial on learning with Bayesian networks. In: Innovations in Bayesian Networks: Theory and Applications, pp. 33–82 (2008)
17. Hepler, A.B., Dawid, A.P., Leucari, V.: Object-oriented graphical representations of complex patterns of evidence. Law Prob. Risk **6**(1–4), 275–293 (2007)
18. Jeffrey, R.C.: The Logic of Decision. McGraw-Hill Series in Probability and Statistics, 2nd edn. McGraw-Hill, New York; University of Chicago Press, Chicago (1983)
19. Keppens, J.: Argument diagram extraction from evidential Bayesian networks. Artif. Intell. Law **20**(2), 109–143 (2012)
20. Koiter, J.R.: Visualizing inference in Bayesian networks. Master's Thesis, Department of Computer Science, Delft University of Technology (2006)
21. Korb, K.B., Nicholson, A.E.: Bayesian Artificial Intelligence. CRC Press (2010)
22. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles and Techniques. MIT Press, Cambridge, MA (2009)
23. Kuhn, D.: The Skills of Argument. Cambridge University Press, Cambridge (1991)
24. Laurila, M., Mäntyniemi, S., Venesjärvi, R., Lehikoinen, A.: Incorporating stakeholders' values into environmental decision support: a Bayesian Belief Network approach. Sci. Total Environ. **697**, 134026 (2019)
25. Lauritzen, S.L.: Graphical Models, vol. 17. Clarendon Press, Oxford (1996)
26. McConachy, R., Korb, K.B., Zukerman, I.: A Bayesian approach to automating argumentation. In: New Methods in Language Processing and Computational Natural Language Learning (1998)
27. McConachy, R., Zukerman, I.: Dialogue Requirements for Argumentation Systems. Linköping University Electronic Press (1999)
28. Nagarajan, R., Scutari, M., Lèbre, S.: Bayesian Networks in R. Springer **122**, 125–127 (2013)
29. Nyberg, E.P., et al.: BARD: a structured technique for group elicitation of Bayesian networks to support analytic reasoning. Risk Anal. **42**, 1155–1178 (2022)
30. Oaksford, M., Hahn, U.: A Bayesian approach to the argument from ignorance. Can. J. Exp. Psychol. **58**, 121–131 (2004)
31. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Elsevier (1988)
32. Pettigrew, R.: Accuracy and the Laws of Credence. Oxford University Press, Oxford (2016)
33. Pilditch, T.D., Fries, A., Lagnado, D.A.: Deception in evidential reasoning: willful deceit or honest mistake? In: Proceedings of the CogSci, pp. 931–937 (2019)
34. Rescher, N.: Dialectics: A Controversy-Oriented Approach to the Theory of Knowledge. SUNY Press, Albany (1977)

35. Rehder, B., Waldmann, M.R.: Failures of explaining away and screening off in described versus experienced causal learning scenarios. Memory Cognit. **45**, 245–260 (2017)
36. Rudin, C.: Stop explaining black-box machine learning models for high-stakes decisions and use interpretable models instead. Nat. Mach. Intell. **1**(5), 206–215 (2019)
37. Scutari, M., Vitolo, C., Tucker, A.: Learning Bayesian networks from big data with greedy search: computational complexity and efficient implementation. Stat. Comput. **29**, 1095–1108 (2019)
38. Sevilla, J.: Finding, scoring, and explaining arguments in Bayesian networks. arXiv preprint arXiv:2112.00799 (2021)
39. Tesic, M., Hahn, U.: Explanation in AI systems. In: Muggleton, S., Chater, N. (eds.) Human-Like Machine Intelligence. Oxford University Press, Oxford (2021)
40. Timmer, S.T., Meyer, J.-J.C., Prakken, H., Renooij, S., Verheij, B.: A two-phase method for extracting explanatory arguments from Bayesian networks. Int. J. Approx. Reason. **80**, 475–494 (2017)
41. Toulmin, S.E.: The Uses of Argument. Cambridge University Press, Cambridge (1958)
42. Walton, D.N.: Informal Logic: A Handbook for Critical Argument. Cambridge University Press, New York (1989)
43. Walton, D.N.: Argument Structure: A Pragmatic Theory. University of Toronto Press, Toronto (1996)
44. Wellman, M.P., Henrion, M.: Explaining 'explaining away'. IEEE Trans. Pattern Anal. Mach. Intell. **15**(3), 287–292 (1993)
45. Woods, J.: The Death of Argument: Fallacies in Agent-Based Reasoning. Springer, Dordrecht (2004)

Enhancing Argument Generation Using Bayesian Networks

Yuan Cao[1,4(✉)], Rafael Fuchs[2,3], and Anita Keshmirian[1,2,5]

[1] Fraunhofer-Institut für Kognitive Systeme IKS, Munich, Germany
yuan.cao@mein.gmx
[2] Munich Center for Mathematical Philosophy (MCMP) - LMU, Munich, Germany
Rafael.Fuchs@campus.lmu.de
[3] Graduate School of Systemic Neuroscience (GSN) - LMU, Munich, Germany
[4] Technical University of Munich, Munich, Germany
[5] Forward College, Berlin, Germany

Abstract. In this paper, we examine algorithms that utilize factor graphs from Bayesian Belief Networks to generate and evaluate arguments. We assess their strengths and weaknesses, which leads to the creation of our improved algorithm that rectifies the issues that we identified. Our approach includes applying the original and modified algorithms to previously known networks to pose challenges in generating robust arguments for humans and computers. Our findings reveal significant improvements in the creation of more robust arguments. Moreover, we delve into the dynamics of argument interaction, offering detailed insight into the algorithms' practical efficacy.

Keywords: Argument Strength · Bayesian Belief Network · Argument Generation

1 Introduction

Argumentation is central to collective reasoning, informed decision-making, and decision articulation within collaborative contexts. Yet uncertainty pervades decision-making in real life: in a medical setting, doctors often face uncertain scenarios where they must make critical decisions based on incomplete information. Investment decisions are fraught with uncertainty in finance. In the judiciary, verdicts by judges and juries frequently rely on evidence that lacks absolute certainty. Therefore, realistically applicable argumentation theory must be able to cope with reasoning under uncertainty.

Supported by Deutsche Forschungsgemeinschaft (DFG) - Project number 455912038 and the Bavarian Ministry for Economic Affairs, Regional Development and Energy as part of a project to support the thematic development of the Fraunhofer Institute for Cognitive Systems IKS

Y. Cao, R. Fuchs and A. Keshmirian—Equally contributed to this work.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 253–265, 2024.
https://doi.org/10.1007/978-3-031-63536-6_15

The necessity to navigate the complexities of decision-making under uncertainty has sparked significant interest in developing algorithms that facilitate probabilistic reasoning. Such algorithms could enhance the explainability of expert systems, particularly those utilizing Bayesian Belief Networks (BBNs).

BBNs are graphical tools for modeling probabilistic dependencies between variables and facilitating reasoning under uncertainty [4,5]. The use of BBNs to provide explanations in real-world settings faces challenges because of the complexity involved, with many variables and detailed interactions. Developing explanations through the extraction of arguments underscores the need for an argumentation theory that effectively navigates uncertainty and is comprehensible to experts and non-experts.

One method to elucidate BBNs' decisions involves distilling complex arguments into more straightforward, comprehensible segments. However, simplifying arguments presents a dichotomy: while disassembling complex arguments into simpler components enhances transparency and comprehensibility, it risks oversimplification, where the interconnected nature of premises is pivotal. Hence, there is a fundamental trade-off: make the representation of an argument as straightforward as possible while maintaining a sufficient level of accuracy concerning the underlying probabilistic reasoning structure. This balance - streamlining argument representation without compromising the integrity of the underlying probabilistic logic - is at the heart of our paper.

In Sect. 2, we motivate the question of independent arguments and introduce an algorithm by Sevilla [7] that uses factor graphs to extract arguments from BBNs, which also gives a useful criterion for independent arguments.

In Sect. 3, we identify some problems in Sevilla's algorithm using a scenario known as "The Spider" [6] to assess the performance of algorithms in providing explanations. The Spider case is notable for previously testing both human and artificial agents with its complex scenario, frequently uncovering instances of less-than-ideal reasoning. [2]. We propose our own improvements to the algorithm by showing its enhanced reasoning. Finally, we present our improved version results and demonstrate the threshold's merits for independent arguments in the factor graph approach.

2 The Question of Independent Arguments

In this section, we look at probabilistic argumentation, explicitly examining how arguments depend on (or are independent of) each other. We focus on extracting arguments from BBNs using factor graphs. First, we briefly introduce factor graphs. Following this, we present a detailed overview of a specific algorithm, as proposed by Sevilla [7], explaining its methodology in the field of probabilistic argumentation. Apart from Sevilla's work, the work on extracting arguments from BBNs is scarce. Other algorithms rely on graphical methods (e.g., [8,9]), but none use factor graphs. Since Sevilla's factor graph approach is novel in this respect, we aim to explore its potential and the power of its criterion for argument independence.

2.1 Factor Graphs

In probabilistic argumentation, it is essential to identify when arguments are independent, as this clarity helps to understand each argument's role in a complex discussion. Factor graphs, which build upon the ideas of BBNs, provide a clear framework for mapping and studying the parts and behavior of arguments. This approach is especially useful when the argumentation process can be simplified into smaller, more manageable functions, each concerning a specific set of variables.

Technically, factor graphs are a type of graphical model used in probability theory and statistical modeling to represent the factorization of a function. Consider a probability distribution $P(X_1, X_2, \ldots, X_n)$ over n random variables. This distribution can be factorized as:

$$P(X_1, X_2, \ldots, X_n) = \prod_{k=1}^{K} f_k(S_k)$$

where $f_k(S_k)$ represents a factor over a subset of variables, and K is the number of factors. Graphically, this factorization is represented as a bipartite graph with variable nodes (X_i) and factor nodes (f_k). An edge is drawn between a variable node and a factor node if the variable is in the subset for that factor. For details, see [1,4].

When using factor graphs for argument extraction, variable nodes can represent components of an argument, such as claims, evidence, counterarguments, and assumptions, whereas factor nodes represent inference rules. Each element plays a distinct role in the structure of the argument. Factors represent the probabilistic relationships between these components, e.g., a factor might represent the strength of evidence supporting a claim or the impact of a counterargument on the overall argument's validity. Using probabilistic models, the factor graph can accommodate uncertainties and variabilities inherent in arguments, including assessing the likelihood of a claim's validity based on the available evidence.

2.2 Overview of the Factor-Graph-Approach Proposed by J. Sevilla

The algorithm constructs a factor graph from a BBN as follows[1]. It creates variable nodes for each variable from the BBN and factors representing the conditional probability tables. Connections between variable nodes and their respective factors are established in these conditional probabilities. To calculate and update joint probability distributions in the factor graph, the message passing algorithm [4] is used.

In preparation for message passing, observation nodes are set to lopsided factors (i.e., zero or one) for the initialization phase, reflecting known states with a probability of one and all other states with zero probability. Other nodes are initialized with constant factors, assuming a uniform distribution. Once the factor

[1] For sources containing a pseudo-code and the technical implementation, see material in Appendix A.

graph is established, the algorithm implements the message-passing algorithm to calculate the flow of messages across the graph.

Effects and Strength of an Argument: This approach represents arguments as directed acyclic graphs over the factor graph. An argument, for example, is shown in Fig. 4. It comprises nodes and factors ranging from observation to the target node. The influence of each inference step in an argument is called *Step Effect* and is defined by how a preceding node impacts the subsequent node. More specifically, the argument's premises (variable nodes) are multiplied with their factor node (inference rule) as per the message-passing algorithm, and the result is normalized by dividing by the factor itself. This division distinguishes new information (Δ) and inherent data in the conditional probability table (ϕ).

The cumulative effect of an argument is calculated by multiplying the effects of all parent factors through the recursive application of the step effect. Finally, the strength of an argument is measured by the logarithmic odds of its effect supporting the outcome. This provides a real-valued metric that indicates the argument's direction (support or opposition) and magnitude (strength).

Argument Independence: Determining argument independence involves assessing if the combined effect of multiple arguments equals the product of their individual effects. Arguments are independent if their effect's discrepancy falls within a predefined threshold. This is measured as the maximum absolute difference in log odds between the factors, represented by the equation:

$$\text{Factor Distance}(\phi_1, \phi_2) = \max \left| \log \frac{(\phi_1/\phi_2(t_0))\,(t_o)}{\text{Average}_{t \neq t_o}\,(\phi_1/\phi_2)\,(t)} \right|,$$

where $(\phi_1/\phi_2)(t_0)$ is the probability ratio $\phi_1(t_0)$ to $\phi_(t_0)$ (i.e. the probability that variable T takes value t_0 given ϕ_1 vs that probability given ϕ_2), which is compared to the average of all values t of T such that $t \neq t_0$ (see pseudo-code in Appendix A).

Finding All Arguments: The algorithm's objective is to identify a set of relevant and independent arguments that elucidate the network's outcome based on given premises and a target. It begins by identifying simple arguments[2]) from each evidence node to the target, excluding paths passing through another evidence node. The algorithm then iteratively combines these simple arguments into more complex ones, checking for their potential breakdown into independent combinations. Two thresholds are set to accommodate larger BBNs: one for the length of simple paths (from one premise node to the query node) and another for the number of these simple paths to be combined. Finally, dependent arguments are amalgamated, and all arguments are ordered by their absolute strength.

[2] an argument is simple if it cannot be broken down into a union of distinct sub-arguments, [7, p. 6].

Explaining Arguments: Natural language explanations of arguments are generated by tracing the nodes each simple argument passes through. The outcome is determined based on the evidence favored by the message-passing algorithm's results.

3 Testing and Improving the Factor Graph Algorithm

In this section, we identify some problems in Sevilla's algorithm that lead to incorrect results in an application scenario ("The Spider") we used to test it. We then propose improvements and show how the improved algorithm yields better outcomes. Finally, we test different threshold levels for independent arguments.

3.1 Overview of the BARD Project and "the Spider" Problem

The BARD project [2,3] sets out to establish an overarching framework leveraging BBNs to advance argumentation. This initiative mainly tailors decision scenarios to underscore the complexities and challenges faced in decision-making endeavors mediated by BBNs, focusing on navigating through evidence conflicts, gauging source reliability, and encapsulating uncertainty to ensure clarity and comprehension.

Our research focuses on the "The Spider" problem presented in the BARD project, as described by Pilditch (2019) [6]. This scenario serves as a testing ground for dealing with misleading information sources.

In this exercise, participants assume the role of intelligence analysts on the hunt for a notorious foreign spy, known as "The Spider," suspected to be hiding in a facility located in a neutral country. The primary objective is to gather additional intelligence to determine the necessity of a covert operation to capture the Spider. Initial reports from agents Emerson and Quinn place the Spider within the facility, with both agents acclaimed for their high reliability (characterized by low false-positive and false-negative rates). However, emerging telephone records cast suspicion on Emerson and Quinn's loyalty, insinuating they might collaborate with the Spider. On the other hand, the records might also be forged: the Spider's true allies might have created them to spread disinformation. If the records turn out to be authentic, it would mean that Emerson and Quinn consistently report the opposite (i.e., if the Spider is in the facility, they report that he is not, and vice versa).

Finally, Winter, a communication analyst known for her meticulousness (almost zero false positives), confirms the Spider's presence through surveillance data. Trustworthy field agent Sawyer and local witness Alpha echo this claim. The structure of this scenario is visualized in the BBN shown in Fig. 1 (all variables are binary).

The decision-making process in this scenario is challenging due to conflicting reports from Emerson, Quinn, and the other members. In particular, uncertainty regarding the authenticity of the telephone records adds another layer of complexity to the conflict. How should we weigh the highly reliable information of one group reporting negatively against the collective inputs of the other members reporting positively? This dilemma underscores the intricacy of the Spider problem and highlights the need for an effective strategy to resolve such conflicts. We will implement the algorithms for this problem to analyze their reliability.

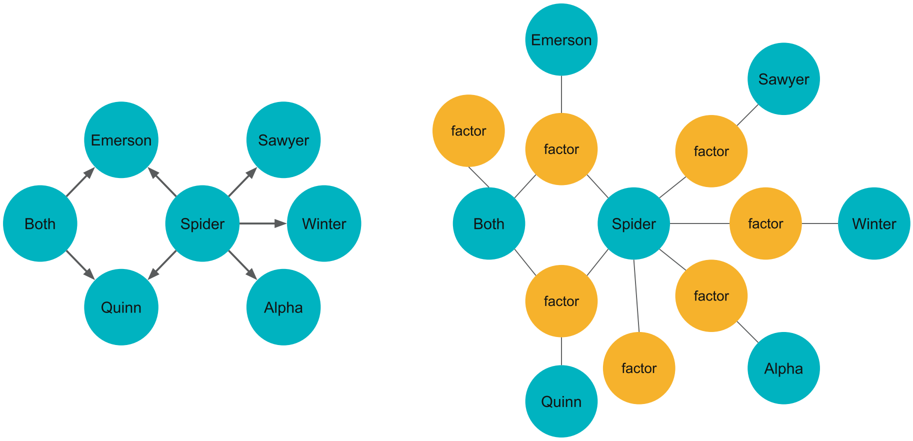


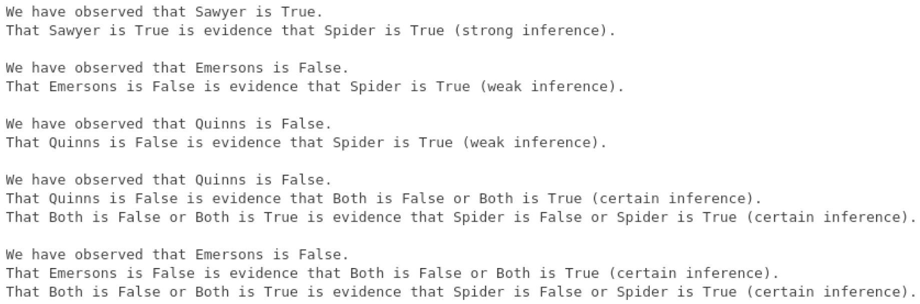
Fig. 1. The structure of the Spider network and its factor graph. Left: the BBN of "The Spider". Right: The factor graph of "The Spider" network. The blue nodes represent the nodes in the BBN, and the orange nodes represent factors. (Color figure online)

3.2 Results with the Original Algorithm

In this section, we apply Sevilla's original algorithm to "The Spider" problem, addressing a fundamental question: based on your evidence, "the Spider is not in the facility" from Emerson and Quinn and "the Spider is in the facility" from Sawyer, what do you believe the probability is of "The Spider" is in the facility? Additionally, we adjust the threshold settings to explore the interactions between different arguments.

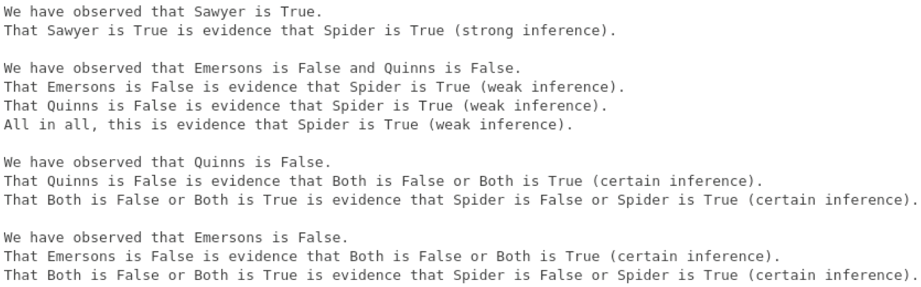
Each paragraph in Fig. 2 and Fig. 3 is an argument. For instance, the structure of the first argument in Fig. 2 is from Sawyer to "The Spider" as shown in Fig. 4. The arguments favor "The Spider" being in the facility (Spider is true) or neutral (Spider is true or Spider is false). This means that taken together, the arguments of this algorithm suggest "The Spider" is in the facility when it is known that Emerson and Quinn report the absence of "The Spider" and Sawyer reports the presence of "The Spider".

As depicted in Fig. 2, the default threshold condition results in a clear separation of all arguments. Upon reducing the threshold value, we observe that arguments are identified as being interdependent. Figure 3 illustrates an interaction between the arguments originating from Quinn and Emerson towards Spider, which is a notable deviation from their previously independent status shown in Fig. 2. The threshold deciding the interaction level of arguments is user-defined



```
We have observed that Sawyer is True.
That Sawyer is True is evidence that Spider is True (strong inference).

We have observed that Emersons is False.
That Emersons is False is evidence that Spider is True (weak inference).

We have observed that Quinns is False.
That Quinns is False is evidence that Spider is True (weak inference).

We have observed that Quinns is False.
That Quinns is False is evidence that Both is False or Both is True (certain inference).
That Both is False or Both is True is evidence that Spider is False or Spider is True (certain inference).

We have observed that Emersons is False.
That Emersons is False is evidence that Both is False or Both is True (certain inference).
That Both is False or Both is True is evidence that Spider is False or Spider is True (certain inference).
```

Fig. 2. Results from the original algorithm with default threshold = 0.1.



```
We have observed that Sawyer is True.
That Sawyer is True is evidence that Spider is True (strong inference).

We have observed that Emersons is False and Quinns is False.
That Emersons is False is evidence that Spider is True (weak inference).
That Quinns is False is evidence that Spider is True (weak inference).
All in all, this is evidence that Spider is True (weak inference).

We have observed that Quinns is False.
That Quinns is False is evidence that Both is False or Both is True (certain inference).
That Both is False or Both is True is evidence that Spider is False or Spider is True (certain inference).

We have observed that Emersons is False.
That Emersons is False is evidence that Both is False or Both is True (certain inference).
That Both is False or Both is True is evidence that Spider is False or Spider is True (certain inference).
```

Fig. 3. Results from the original algorithm with threshold $= 2 \times 10^{-16}$.

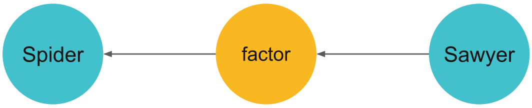


Fig. 4. The first argument in Fig. 2. The direction is from the observation to the query node.

and can be adjusted based on specific situations. The optimal threshold varies depending on the scenario.

3.3 Diagnosis and Solution Proposal

Here, we present our in-depth exploration of the algorithm's technical difficulties and shortcomings. We provide a comprehensive analysis of their causes and effects. Following this analysis, we propose targeted solutions and enhancements to improve the algorithm's accuracy and reliability.

Ignorance of Prior Probability. The initialization of the nodes without information assumes a uniform distribution, which leads to the wrong calculation of the probability marginalization of the outcome. In Fig. 2, Quinn's report of the Spider's absence paradoxically suggests the Spider's presence, contrasting our initial expectations. We anticipate that if Quinn reports the absence of the Spider, it would significantly increase the likelihood of its absence, considering the low propensity to be league with the Spider. To rectify this, we propose changing the initialization of nodes, except for evidence nodes, to reflect their prior probabilities.

Certain Inference. When distinct node states are assigned equivalent probabilities, the algorithm returns a "certain inference." However, this might be misleading about a definitive node's state, which is not the case. To address this semantic inconsistency, we propose renaming this outcome "equal effect inference."

D-Separation Detection Deficiency. The algorithm is unable to identify d-separation structures: two (non-empty) sets of nodes X, Y are d-separated by another (possibly empty) set of nodes Z, if and only if every path from a node $x \in X$ to a node $y \in Y$ is blocked. A path $x_i \rightarrow v \rightarrow ... \rightarrow y$ is blocked by Z iff for every node w on the path one of the following two holds:

1. the path's edges do not meet head-to-head in w and $w \in Z$, or
2. the edges meet head-head in w and $w \notin Z$ **and** none of w's descendant are in Z.

D-separation identifies conditional independence relations between nodes in a Bayes net. Our results indicate that an effect exists between d-separated nodes. We adapted the algorithm to evaluate d-separation between nodes for every step of the argument process. An identification of d-separation signifies that the argument does not affect the target node.

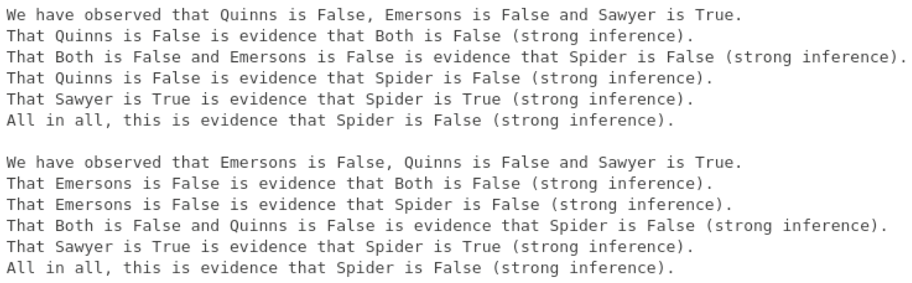
Uncertain Equivalence Between Node Value and Step Effect. In each step, the value of the step target node equals the step effect when moving from parent to child. Conversely, from child to parent, the value equals the step effect times the parent's prior probability. This distinction arises because the step effect represents $P(\text{child}|\text{parent})$. When calculating $P(\text{parent}|\text{child})$, it equals $P(\text{child}|\text{parent}) * P(\text{parent})/P(\text{child})$ according to Bayes rule. By first determining the direction of the effect, we increase the precision of our effect and strength calculations.

Table 1. Conclusion of the improvements

Dimensions	The original algorithm	Our improved algorithm
Initialization of unobserved nodes	uniform distribution	prior probability
Explanation in words	certain inference	equal effect inference
D-separation	Non D-separated detection	D-separated detection addition
Linking the possibility and step effect	Equivalence	Considerations of node relationship

To summarise, our improvements are listed in Table 1.

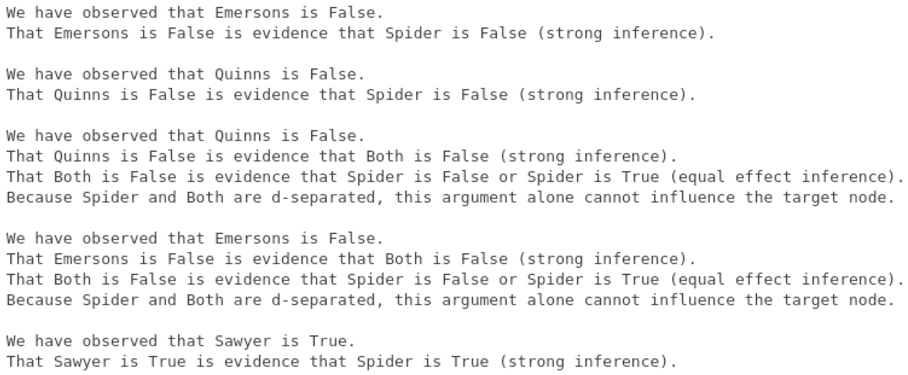
3.4 Results of the Improved Version



```
We have observed that Quinns is False, Emersons is False and Sawyer is True.
That Quinns is False is evidence that Both is False (strong inference).
That Both is False and Emersons is False is evidence that Spider is False (strong inference).
That Quinns is False is evidence that Spider is False (strong inference).
That Sawyer is True is evidence that Spider is True (strong inference).
All in all, this is evidence that Spider is False (strong inference).

We have observed that Emersons is False, Quinns is False and Sawyer is True.
That Emersons is False is evidence that Both is False (strong inference).
That Emersons is False is evidence that Spider is False (strong inference).
That Both is False and Quinns is False is evidence that Spider is False (strong inference).
That Sawyer is True is evidence that Spider is True (strong inference).
All in all, this is evidence that Spider is False (strong inference).
```

Fig. 5. Results from our updated algorithm with default threshold = 0.1.

After implementing our enhanced algorithm to revisit “The Spider” case, we observed that the outcomes were significantly more plausible than the original results. When Emerson or Quinn reports the argument, arguments are identified as independent with an elevated threshold. This outcome is consistent with their established reliability and the low likelihood of them being allied with the Spider. The outcomes presented in Fig. 5 demonstrate a merging of arguments under the standard threshold. Conversely, Fig. 6 shows that the arguments are identified as independent with an elevated threshold.



```
We have observed that Emersons is False.
That Emersons is False is evidence that Spider is False (strong inference).

We have observed that Quinns is False.
That Quinns is False is evidence that Spider is False (strong inference).

We have observed that Quinns is False.
That Quinns is False is evidence that Both is False (strong inference).
That Both is False is evidence that Spider is False or Spider is True (equal effect inference).
Because Spider and Both are d-separated, this argument alone cannot influence the target node.

We have observed that Emersons is False.
That Emersons is False is evidence that Both is False (strong inference).
That Both is False is evidence that Spider is False or Spider is True (equal effect inference).
Because Spider and Both are d-separated, this argument alone cannot influence the target node.

We have observed that Sawyer is True.
That Sawyer is True is evidence that Spider is True (strong inference).
```

Fig. 6. Results from our updated algorithm with threshold = 6.

Figure 6 further showcases the ability of the algorithm to detect d-separation. Analyzing an individual argument with Quinn leading to Spider via Both, the nodes Both and Spider are d-separated within the Both ← Emerson → Spider collider structure. The impact of Quinn on Spider is interrupted in this sequence. The algorithm detects the d-separation and informs users that this particular type of argument does not influence the target node.

4 Limitation and Future Work

This paper identifies and addresses key areas for enhancement within the factor graph-based approach to the algorithmic generation and evaluation of arguments. We have introduced modifications that considerably bolster reasoning capabilities. Our preliminary research, centered on the exemplary use of a complex and challenging Bayesian Belief Network ("The Spider"), has illuminated promising avenues for refining reasoning strategies. Despite these advancements, there remains substantial scope for future research to validate these algorithmic improvements across a more varied array of scenarios and Bayesian Belief Networks (BBNs), thus underlining their widespread applicability and efficacy.

Through this exploration, we enhance reasoning capabilities and underscore the significance of setting a threshold for independent arguments within the factor graph framework. This work establishes a solid foundation for further investigation into the algorithm's operational effectiveness. Building upon this foundation, we aim to extend our analysis to a wider range of BBNs. This endeavor is motivated by our goal to affirm the universality and practical utility of the proposed algorithmic enhancements.

Moreover, future research is crucial to build upon our findings through empirical evaluation. This subsequent research phase will compare the human understanding and evaluation of the algorithm's arguments against its actual performance. By incorporating a more extensive set of examples and applying quantitative accuracy metrics, we aim to solidify the evidence supporting our claims of improved algorithmic performance. This approach addresses the limitations identified and deepens our comprehension of how these algorithmic enhancements can significantly enhance human reasoning processes in the face of uncertainty.

5 Conclusion

This paper pinpoints and tackles crucial improvement opportunities within the factor-graph-based approach to generating and evaluating arguments using Bayesian Belief Networks (BBNs). We have implemented changes that strengthen the reasoning abilities of an exemplary algorithm that uses factor graphs.

Refining Sevilla's algorithm, we demonstrated that meaningful argument extractions from BBNs are possible within this approach. We especially noted the utility of establishing a threshold for independent arguments. This feature, in particular, showcases the potential for more precise and nuanced argumentation within complex probabilistic models.

Acknowledgements. We thank the Deutsche Forschungsgemeinschaft (DFG) and the Fraunhofer Institute for Cognitive Systems IKS for supporting our research. We thank Dr. Narges Ahmidi and Alireza Zamanian for their insightful suggestions and valuable feedback on this paper. We would also like to thank anonymous reviewers for their useful comments.

A Appendix

Our enhancements to this algorithm, based on Sevilla's packages and the PGMPy open-source software for computing Bayesian networks, can be viewed in detail at this link[3]. To help you understand our enhancements better, we have attached pseudo-code for the relevant parts.

For the initialization of the unobserved nodes, they are under a uniform distribution initialization:

```
for node in model:
    if node in evidence:
    node[observed_state] = 1
    node[other_states] = 0
    else:
    node[states] = uniform distribution
```

[3] https://github.com/yuancao-git/factor_graph_algorithm.git.

We modify it to make use of prior probabilities:

```
for node in model:
    if node in evidence:
    node[observed_state] = 1
    node[other_states] = 0
    else:
    prior = model.prior_calculation(node)
    node[states] = prior
```

We change how we calculate the argument's strength to make the explanations more precise. The outcomes are altered from "certain inference":

```
query_node = [v1, v2, ..., vn] (set V)
if query_node has one maximum vm:
    argument_strength = vm/sum(V out of vm)
if query_node has several maximums vm1,...,vmm:
    choose all these states vm1,...,vmm (set Vm)
    argument_strength=sum(vm1,...,vmm)/sum(V out of Vm)
```

to the "equal effect inference":

```
query_node = [v1, v2, ..., vn] (set V)
if query_node has one maximum vm:
    argument_strength = vm/sum(V out of vm)
if query_node has several maximums vm1,...,vmm:
    choose all these states vm1,...,vmm (set Vm)
    argument_strength = vm1 / sum (V out of Vm)
    display ‘‘equal effect inference’’
```

Before calculating the argument strength, we add D-separation detection to check if there is influence between nodes:

```
for argument in all_arguments:
    for step in argument:
    sub_BBN = to_BBN(argument, step)
    d_separation =
        d_separated(sub_BBN, evidence_in_argument)
    if d_separation == True:
        ‘‘no influence’’
    if d_separation == False:
        calculate argument_strength
```

References

1. Acar, U.A., Ihler, A., Mettu, R., Sümer, O.: Adaptive Bayesian inference. Neural Information Processing Systems (NIPS), vol. 10, pp. 2981562–2981743 (2007)
2. Cruz, N., et al.: Widening access to Bayesian problem-solving. Front. Psychol. **11**, 660 (2020)
3. Dewitt, S., Lagnado, D., Fenton, N.: Updating prior beliefs based on ambiguous evidence. In: COGSCI2018: Changing Minds, pp. 2047-2052 (2018)
4. Koller, D., Friedman, N.: Probabilistic Graphical Models: Principles And Techniques. MIT Press, Cambridge, MA (2009)
5. Lauritzen, S.L.: Graphical Models, vol. 17. Clarendon Press, Oxford (1996)
6. Pilditch, T.D., Fries, A., Lagnado, D.A.: Deception in evidential reasoning: Willful deceit or honest mistake? In Proceedings of the CogSci, pp. 931-937 (2019)
7. Sevilla, J.: Finding, Scoring, and Explaining Arguments in Bayesian Networks. arXiv preprint arXiv:2112.00799. https://arxiv.org/abs/2112.00799 (2021)
8. Timmer, S.T., Meyer, J.-J.C., Prakken, H., Renooij, S., Verheij, B.: A two-phase method for extracting explanatory arguments from Bayesian networks. Int. J. Approximate Reasoning **80**, 475–494 (2017)
9. Keppens, J.: Argument diagram extraction from evidential Bayesian networks. Artif. Intell. Law **20**(2), 109–143 (2012)

"Do Not Disturb My Circles!" Identifying the Type of Counterfactual at Hand *(Short Paper)*

Moritz Willig[1](✉), Matej Zečević[1], and Kristian Kersting[1,2,3,4]

[1] Computer Science Department, Technical University of Darmstadt,
Darmstadt, Germany
moritz.willig@cs.tu-darmstadt.de

[2] Centre for Cognitive Science, Technical University of Darmstadt,
Darmstadt, Germany

[3] Hessian Center for Artificial Intelligence (hessian.AI), Darmstadt, Germany

[4] German Research Center for Artificial Intelligence (DFKI),
Darmstadt, Germany

Abstract. When the phenomena of interest are in need of explanation, we are often in search of the underlying root causes. Causal inference provides tools for identifying these root causes—by performing interventions on suitably chosen variables we can observe down-stream effects in the outcome variable of interest. On the other hand, argumentation as an approach of attributing observed outcomes to specific factors, naturally lends itself as a tool for determining the most plausible explanation. We can further improve the robustness of such explanations by measuring their likelihood within a mutually agreed-upon causal model. For this, typically one of in-principle two distinct types of counterfactual explanations is used: interventional counterfactuals, which treat changes as deliberate interventions to the causal system, and backtracking counterfactuals, which attribute changes exclusively to exogenous factors. Although both frameworks share the common goal of inferring true causal factors, they fundamentally differ in their conception of counterfactuals. Here, we present the first approach that decides when to expect interventional and when to opt for backtracking counterfactuals.

Keywords: Explanations · Causality · Interventions · Backtracking

1 Introduction

Dating back to the times of Aristotle, causality as a concept is intimately linked with human reasoning and the formation of arguments (Evans, 1959; Falcon, 2006). When trying to find the truth over a topic by exchanging arguments, we rely on causal relations to ground our claims within the realm of observed and already agreed-upon knowledge (Hume, 1896). In summary, causality is the key

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 266–275, 2024.
https://doi.org/10.1007/978-3-031-63536-6_16

factor that lets us distinguish between mere coincidences and true relations of cause and effect (Mackie, 1980; Aldrich, 1995).

While day-to-day arguments might (or might rather not) involve explicit notions of causality, we assume in this paper that certain kinds of mechanisms dictate the unfolding of events in our everyday lives. While, in principle, arbitrary events could alter the unfolding of things, we assume that there exists a 'natural' –that is, an *unintervened*– unfolding of events. When trying to provide arguments about possible alternative outcomes, one might try to come up with a possible 'counterfactual' unfolding of events that only requires small deviations from the, otherwise, natural unfolding. This heuristic is based on the assumption that explanations adhere to previous observations and past experiences. In this paper, we argue that arguments should stay close to previous experience and only be abandoned in the light of new evidence. Such situations usually only happen when explicit information about such interventions taking place is obtained or observations deviate from expectations to the extent that assuming interventions to be the underlying cause is inevitable.

In the following sections, we discuss the use of different counterfactual explanation methods within graphical causal models. More formally, we utilize the Pearlian notion of causality (Pearl, 2009) to reason about underlying causal relations. While we derive our reasoning method with the formalism of Pearlian causality, we want to point out that several methods exist to transform arguments to causal Bayesian networks, and vice versa to extract arguments from those (Bex et al., 2016; Timmer et al., 2017; Wieten et al., 2019).

Contributions. In the following we briefly sketch a possible application of our idea in a potential case of a court hearing. We discuss the benefits and shortcomings of classical interventional and backtracking counterfactuals. Both approaches aim to generate arguments for hypothetical and/or counterfactual scenarios. We propose an algorithm to infer the most plausible explanation from a natural unfolding of a system and fall back to interventional explanations when needed. Lastly, we propose the use of infinitesimal probabilities in causal models as a way of comparing explanations across multiple interventional distributions.

1.1 Introductory Example

To the best of our knowledge, counterfactual causal reasoning has not yet been applied to the field of formal argumentation (as, for example, summarized by Baroni et al. (2011)). Such approaches might be particularly useful to a confined set of settings, where one tries to argue over hypothetical –*counterfactual*– scenarios. We will now present a hypothetical applied example to better motivate the assumptions made during the following sections.

Consider the hypothetical scenario of a debate during a court hearing on whether or not some store employee could have helped an injured customer. While the fact that the employee did not help is undisputed, a defense attorney might try to argue that all attempts to provide such help would have also been bound to fail. Therefore, all arguments remain in reasoning about non-observable *counterfactual* outcomes. In such a scenario, a successful defense would naturally try to derive zero probability for all possible positive (in terms of successfully helping the customer) outcomes. The opposing prosecutor would try to come up with feasible counterarguments. A possible line of argument could go as follows:

Argument: Even if the employee had been willing to help, no medicine was available. *Attack*: Assuming that medicine would have become available, the employee should have started with the emergency procedure.

Argument: The employee did not receive proper training to start the procedure. *Attack*: Proper training was offered to all employees. *And so on...*

Every argument tries to reduce the probability of a successful help outcome to zero. In such cases, the only remaining attack is to assume that some latent factor (e.g. presence of medicine, proper training, ...) could have been set to another value than the one that is claimed. Given the benefit of the doubt, every such assumption required to accuse the employee weakens the indictment and lowers the chance of conviction. The total number of such necessary assumptions is likely to influence the final court ruling. In the following, we will capture this notion via a preorder on argument preference in Sect. 3.2.

2 Preliminaries and Related work

In general, we write indexed sets of variables in bold upper-case $\mathbf{X}$ and their values in lower-case $\mathbf{x}$. Single variables and their values are written in normal style (X, x). Specific elements of a tuple are indicated by a subscript index X_i. Probability distributions of a variable X or a tuple $\mathbf{X}$ of variables are denoted by P_X and $\mathrm{P}_{\mathbf{X}}$ respectively.

Structural Causal Models. Structural Causal Models (SCM) provide a framework to formalize a notion of causality via graphical models (Pearl, 2009). They can be expressed as structural equation models without affecting expressiveness (Rubenstein et al., 2017). We adopt a slightly modified definition of SCM modeling an explicit set of allowed interventions, similar to earlier works of Rubenstein et al. (2017); Beckers and Halpern (2019); Willig et al. (2023).

Definition 1. *A structural causal model is a tuple $\mathcal{M} = (\mathbf{V}, \mathbf{U}, \mathbf{F}, \mathcal{I}, \mathrm{P}_{\mathbf{U}})$ forming a directed acyclic graph $\mathcal{G}$ over the indexed set of variables $\mathbf{X} = \{X_1, \ldots, X_K\}$ taking values in $\boldsymbol{\mathcal{X}} = \prod_{k\in\{1\ldots K\}} \mathcal{X}_k$ subject to a strict partial order $<_{\mathbf{X}}$ over $\mathbf{X}$, where*

- $\mathbf{V} = \{X_1, \ldots, X_N\} \subseteq \mathbf{X}, N \leq K$ *is the indexed set of endogenous variables.*
- $\mathbf{U} = \mathbf{X} \setminus \mathbf{V} = \{X_{N+1}, \ldots, X_K\}$ *is the indexed set of exogenous variables.*
- $\mathbf{F}$ *is the indexed set of deterministic structural equations, $V_i := f_i(\mathbf{X}')$, where the parents are $\mathbf{X}' \subseteq \{X_j \in \mathbf{X} \,|\, X_j <_{\mathbf{X}} V_i\}$.*
- $\mathcal{I} \subseteq \{\{I_{i,d_i} \,|\, i \in \mathbf{i}, d_i \in \mathbf{d}\}_{\mathbf{i}\subseteq\{1\ldots N\}}\}_{\mathbf{d}\in\boldsymbol{\mathcal{J}}}$ *where $\boldsymbol{\mathcal{J}}$ is the set of possible (generally unknown) joint distributions $\mathbf{d}$ on $\boldsymbol{\mathcal{X}}$, I_{i,d_i} indicates an intervention $do(X_i \sim d_i)$, where the value of X_i is sampled from the i-th marginal distribution of $\mathbf{d}$. We write arbitrary sets of interventions on $\mathbf{X}' \subseteq \mathbf{X}$ as $\mathbf{I}_{\mathbf{X}'} \in \mathcal{I}$.*

– $\mathrm{P}_{\mathbf{U}}$ *is the probability distribution over* $\mathbf{U}$.

By construction, at most one intervention on any specific variable is to be included in any intervention set $\mathbf{I} \in \mathcal{I}$. When $\boldsymbol{\mathcal{J}}$ is defined to equal $\delta(\boldsymbol{\mathcal{X}})$ (with $\delta(\boldsymbol{\mathcal{X}})$ being the set of all possible Dirac distributions over $\boldsymbol{\mathcal{X}}$), $\mathcal{I}$ models sets of atomic interventions. An atomic intervention on a single variable $do(X_i \sim \delta(x_i'))$ places all probability mass on a single value x_i'. Consequently, the unintervened f_i can be replaced by the constant assignment $X_i := x_i'$ and we write $do(X_i = x_i')$, and I_{i,x_i}, respectively. Every $\mathcal{M}$ entails a DAG structure $\mathcal{G} = (\mathbf{X}, \mathcal{E})$ consisting of vertices $\mathbf{X}$ and edges $\mathcal{E}$, where a directed edge from X_j to X_i exists if $\exists x_0, x_1 \in \mathcal{X}_j . f_i(\mathbf{x}', x_0) \neq f_i(\mathbf{x}', x_1)$. For every variable X_i we define $\mathrm{ch}(X_i), \mathrm{pa}(X_i)$ and $\mathrm{an}(X_i)$ as the set of direct children, direct parents and ancestors respectively, according to $\mathcal{G}$.[1] Every $\mathcal{M}$ entails an observational distribution $\mathrm{P}_{\mathcal{M}}$[2] by pushing forward $\mathrm{P}_{\mathbf{U}}$ through $\mathbf{F}$. Intervention $do(X_i \sim d_i)$ replace f_i by a function sampling from d_i. As a consequence, $\mathcal{M}$ might entail infinitely many intervened distributions $\mathrm{P}_{\mathcal{M}}^{\mathbf{I}}$, generally preventing us from simultaneously modeling all possible scenarios that might arise during an argument (see Sect. 3.3).

3 Backtracking in Causal Models

In this section, we will briefly review inference for classical 'interventional' counterfactuals of Pearl (2009) and compare them to 'backtracking' counterfactuals of Von Kügelgen et al. (2023). We will then present a scenario where backtracking counterfactuals fail to explain the given evidence and propose an iterative method to remedy the situation. Since the following observational and counterfactual values might be inferred over the same set of variables we denote the corresponding counterfactual quantities with $\square^*$.

Interventional Counterfactuals. We write $\mathrm{P}_{\mathcal{M}}(\mathbf{Y}_{\mathbf{x}^*} = \mathbf{y}^* \,|\, \mathbf{e})$ to express the counterfactual question: "What would be the probability of some $\mathbf{Y} \subseteq \mathbf{V}$ taking values $\mathbf{y}^*$ given some observations (or evidence) $\mathbf{e}$, had variables $\mathbf{X}^* \in \mathbf{X}$ taken values $\mathbf{x}^*$". Given a tuple $\mathbf{e}, \mathbf{x}^*$, classical counterfactual inference is performed in three steps (Pearl, 2009):

Step 1 (abduction): Infer the most probable configuration $\mathbf{u}$ given evidence $\mathbf{e}$ by maximizing $\mathrm{P}(\mathbf{u} \,|\, \mathbf{e})$.

Step 2 (action): Act on the model $\mathcal{M}$ by applying interventions $do(\mathbf{X}^* = \mathbf{x}^*)$. Such that $\mathbf{F}' = \{f_i \in \mathbf{F} \,|\, \nexists I_{j,v_j} \in \mathbf{I} . i = j\} \cup \{X_i := x_i^*\}_{\{x_i^* \in \mathbf{x}^*\}}$ and $\mathcal{M}^{\mathbf{I}} = (\mathbf{V}, \mathbf{U}, \mathbf{F}', \emptyset, \mathrm{P}_{\mathbf{U}})$.

Step 3 (prediction): Compute $\mathrm{P}_{\mathcal{M}}^{\mathbf{I}}(\mathbf{Y} = \mathbf{y}^* \,|\, \mathbf{u})$.

[1] We define $\mathrm{ch}(\mathbf{X}), \mathrm{pa}(\mathbf{X})$ and $\mathrm{an}(\mathbf{X})$ for sets of variables $\mathbf{X}$, as the union of sets obtained by individual variable evaluations, e.g., $\mathrm{pa}(\mathbf{X}) = \bigcup_{X \in \mathbf{X}} \mathrm{pa}(X)$.

[2] In this paper, we always reference distributions with respect to some SCM $\mathcal{M}$. The expression $\mathrm{P}_{\mathcal{M}}$ indicates the distribution induced by $\mathcal{M}$ over the full variable set $\mathbf{X}$.

The most likely counterfactual configuration of variables $\mathbf{Y}$ can then be obtained by searching for an $\mathbf{y}^*$ that maximizes $\mathrm{P}^{\mathbf{I}}_{\mathcal{M}}(\mathbf{Y} = \mathbf{y}^* \mid \mathbf{u})$ in the third step.

Backtracking Counterfactuals. By interpreting the counterfactual quantities $\mathbf{X}_i = \mathbf{x}_i^*$ as interventions, interventional counterfactuals 'detach' the affected variables from the inferred $\mathbf{u}$'s by overwriting their structural equations. Von Kügelgen et al. (2023) try to embed counterfactual values more naturally into the inference framework by fixing $\mathbf{x}^*$ but backtracking without interventions to a counterfactual set $\mathbf{u}^*$ which then entails $\mathbf{x}^*$, but might differ from the $\mathbf{u}$ inferred via observations $\mathbf{e}$. Backtracking counterfactuals preserve the unaltered graph structure by trading it for the induction of a new $\mathbf{u}^*$. Throughout the inference of $\mathbf{y}^*$ the values of $\mathbf{u}, \mathbf{u}^*$ should be kept as close as possible (with regard to some similarity measure) such that $\mathrm{P}(\mathbf{U}, \mathbf{U}^*)$ is maximized.

Comparison. When comparing both approaches, one sees that either the structural equations of $\mathcal{M}$ are altered or a new set of exogenous variables is inferred in order to explain a different outcome. As we want to minimize the usage of interventions and apply them only in cases where no other options are viable, we will consider backtracking counterfactuals as the default technique to infer explanations. Stated more simply: 'If we can explain a counterfactual scenario without the use of external interventions we will do so.' Depending on the specific metric connecting $\mathbf{u}$ and $\mathbf{u}^*$, backtracking counterfactuals might infer a vector $\mathbf{u}^*$ that could be inconsistent with our evidence $\mathbf{e}$. For our goal of obtaining arguments that are coherent with given observations we require all $\mathbf{v}^*$ to take the same values as their corresponding counterparts $\mathbf{v}$ constrained via evidence $\mathbf{e}$. Otherwise, one could always come up with explanations by choosing an arbitrary similarity metric and simply disregard the observed evidence. We will now discuss a scenario where backtracking counterfactuals are unable to explain certain situations and interventions need to be applied.

3.1 When Backtracking is not Enough

Under certain conditions, backtracking will always be able to infer a plausible configuration $\mathbf{u}$ given some observation $\mathbf{e}$. Specifically, this is the case whenever the distribution has full support over $\mathbf{X}$, implying that for any $\mathbf{x} \in \mathbf{X}$ the quantity $P(\mathbf{x})$ is non-zero. Thus, we can always find some $\mathbf{u} \in \mathbf{U}$ for any $\mathbf{e}$ such that $\mathrm{P}_{\mathcal{M}}(\mathbf{u}) \neq 0$. However, there exist a multitude of situations where the SCM won't have full support over the joint domain. For better intuition we reiterate the example given by Von Kügelgen et al. (2023, Remark 4): even for the most simple structural equation $X_0 := U_0; X_1 := X_0$, with U_0, X_0, X_1 being Boolean, we are unable to explain the observation ($X_0 = \texttt{True}, X_1 = \texttt{False}$) via any value of U_0. While this example might seem to be oversimplified, it demonstrates the general problem of the backtracking approach: Whenever there exists a deterministic relation between variables (e.g. $X_1 := X_0$) we are unable to independently set both variables to distinct values using U_0. In such settings, we can always choose some $x_1 \neq x_0$ as evidence and end up with a situation that can not be explained via backtracking.

3.2 Iterative Backup

In the aforementioned case, there is no other choice than resorting back to deploying interventions. However, the number of deployed interventions on the system should be kept minimal. We propose a simple algorithm (c.f. Fig. 1) that is gradually backing up to explanations with higher numbers of interventions in the case that a natural –interventionless– explanation can not be derived from evidence.

Algorithm 1 Iterative Backup Counterfactuals in Structural Causal Models

```
procedure IterativeBacktrack($\mathcal{M}, \mathbf{e}$)
    for i:=0 to N do
        $\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}} \leftarrow \text{argmax}_{\mathbf{I}'_{\bar{\mathbf{V}}} \in \{\mathbf{I} \in \mathcal{I} : |\, \mathbf{I} \,| = i\}} \text{Backtrack}(\mathcal{M}^{\mathbf{I}'_{\bar{\mathbf{V}}}}, \mathbf{e})$
        if $\text{P}_{\mathcal{M}}^{\mathbf{I}_{\bar{\mathbf{V}}}}(\mathbf{x}^*) \neq 0$ then
            return $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}})$
        end if
    end for
end procedure
```

Fig. 1. IterativeBacktrack Algorithm. The algorithm searches through different classes of explanations, starting at the unintervened $\mathscr{I}_0$. In case of not obtaining any suitable explanations the algorithm gradually backs up to $\mathscr{I}_{i>0}$ to search for explanations with higher numbers of interventions. Within the procedure standard backtracking (Backtrack) is performed. However, after the first unintervened iteration it is always applied over an already intervened graph $\mathcal{M}^{\mathbf{I}'_{\bar{\mathbf{V}}}}$.

Order of Preference. To express a preference for a natural unfolding of a system we establish an ordering that prefers explanations with no or fewer interventions over those requiring larger numbers of interventions. We express our preference with the following preorder:

$$(\mathbf{x}^*, \mathbf{I}_{\mathbf{V}}) < (\mathbf{x}^*, \mathbf{I}_{\mathbf{V} \setminus V_i}) < \cdots < (\mathbf{x}^*, \{I_{V_i}, I_{V_j}\})_{i \neq j} < (\mathbf{x}^*, \{I_{V_i}\}) < (\mathbf{x}^*, \emptyset) \quad (1)$$

where each explanation $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}})$ with $\bar{\mathbf{V}} \subseteq \mathbf{V}$ consists of a counterfactual variable assignment $\mathbf{x}^*$ and a set of interventions $\mathbf{I}_{\bar{\mathbf{V}}}$ that leads to a non-zero probability $P_{\mathcal{M}}^{\mathbf{I}}(\mathbf{x}^*)$. By this preorder, one explanation $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}})$ is preferred over another $(\mathbf{x}^*, \mathbf{I}'_{\bar{\mathbf{V}}})$ whenever $|\, \mathbf{I}_{\bar{\mathbf{V}}} \,| < |\, \mathbf{I}'_{\bar{\mathbf{V}}} \,|$. As an immediate consequence, this ordering groups together explanations based on the number of interventions. We define explanation classes $\mathscr{I}_i$ with $i \in \mathbb{N}_0$ and define $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}}) \in \mathscr{I}_i$ iff $|\, \mathbf{I}_{\bar{\mathbf{V}}} \,| = i$. With a slight inaccuracy in expression we also refer to any resulting $\text{P}_{\mathcal{M}}^{\mathbf{I}_{\bar{\mathbf{V}}}}(\mathbf{x}^*)$ as belonging to $\mathscr{I}_i$ whenever the associated $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}}) \in \mathscr{I}_i$ holds. Algorithm 1 searches for the best explanation within each class $\mathscr{I}_i$ by finding the most probable configuration $\mathbf{x}^*$ via classical backtracking counterfactuals while maximizing probability by varying $\mathbf{I}$ under the constraint of $|\, \mathbf{I} \,| = i$. Naturally, the algorithm starts at $\mathbf{I} = \emptyset$ and gradually searches through higher classes of $\mathscr{I}_i$. Whenever there exists no viable backtracking explanation for a given set of interventions the initial assumption of $\mathrm{P}^{\mathbf{I}_{\bar{\mathbf{V}}}}_{\mathcal{M}}(\mathbf{x}^*) \neq 0$ for any $\mathbf{V}$ of the backtracking algorithm is violated. For such cases we assume that the argmax returns an arbitrary $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}})$ whose probability $\mathrm{P}^{\mathbf{I}_{\bar{\mathbf{V}}}}_{\mathcal{M}}(\mathbf{x}^*)$ evaluates to zero.

Trivial Explanation. There always exists at least one explanation with non-zero probability (terminating the algorithm). Assuming that all variables have at least one $x_i \in \mathcal{X}_i$ with $P(X_i = x_i) > 0$. Then there exists at least one configuration that is consistent with the given evidence, that is $(\mathbf{x}^*, \{\mathbf{E} := \mathbf{e}; \mathbf{V} \setminus \mathbf{E} := \mathbf{v}^* \setminus \mathbf{e}\})$ with $\mathbf{x}^*$ arbitrary and every $v^* \in \mathbf{V}$ such that $P(\mathbf{V} = v^*) > 0$. This explanation intervenes on all variables that are not set by the evidence, fully factorizing the SCM into its trivial decomposition $P(\mathbf{V}) = \prod_{V_i \in \mathbf{V}} \mathrm{P}(V_i) = 1$ such that any variable is either determined by evidence or intervention with a probability of one.

3.3 Default Logic

Most of today's causal literature operates under the assumption that the set of variables is fixed upon performing causal inference. We find this assumption particularly difficult in the field of argumentation, where novel arguments might be added dynamically by different parties in order to support their positions. It is possible to model hard interventions via instrumental variables as laid out by Von Kügelgen et al. (2023, Appendix A). A downside of instrumental variables is that these auxiliary variables induce additional complexity to the SCM. While it is easy to attach instrumental variables to any of the original variables, it might be challenging to consider all possible interventions that can be performed on the real-world model under consideration. For these reasons, we would like to incorporate interventions only when required. One possible solution to this problem is the adoption of concepts of default logic. In essence, interventions are disregarded during 'normal operation' and only considered when mandatory. Pearl (1988) and Bochman (2023) discuss possible approaches with regard to causality. Still, it is difficult to quantify and compare probabilities taken from an ever-adapting SCM. Probability values returned from a graph under intervention $\mathbf{I}$ are no longer comparable with regard to some other intervention $\mathbf{I}'$ as the preconditions (specifically the number of interventions) changed, resulting in different underlying distributions.

3.4 Integration of Hyperreals

Modeling all possible scenarios as explicit variables comes with the problem, that we are usually unable to anticipate every possible arbitrary intervention that might occur in the future. To tackle this kind of problem, we induce a search order for Algorithm 1 that guarantees it to stop at the minimal $\mathscr{I}_i$. No matter which explanation $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}})$ is returned, we guarantee that there exists no other explanation $(\mathbf{x}'^*, \mathbf{I}'_{\bar{\mathbf{V}}})$ with fewer interventions such that we adhere to the total ordering of Eq. 1. The main difficulty of this problem, however, stems from the inability to encode our preferred order of $\mathscr{I}_i$ within the probabilities

$\mathrm{P}^{\mathbf{I}_X}_{\mathcal{M}}(\mathbf{x}^*)$ itself. For two explanations $(\mathbf{x}^*, \mathbf{I}_{\bar{\mathbf{V}}}), (\mathbf{x}'^*, \mathbf{I}'_{\bar{\mathbf{V}}})$ of different $\mathscr{I}_i$ with non-zero support, $\mathrm{P}^{\mathbf{I}_X}_{\mathcal{M}}(\mathbf{x}^*)$ and $\mathrm{P}^{\mathbf{I}'_X}_{\mathcal{M}}(\mathbf{x}'^*)$ might be ordered arbitrarily. In this regard, we propose a small trick of introducing infinitesimal quantities ε within the probability estimate of our SCM to make quantities comparable across preference classes.

> **Hyperreal Numbers (informal).** We utilize the concept of hyperreal numbers $^*\mathbb{R}$ as an extension of the real numbers $\mathbb{R}$ (Robinson, 2016). For this, we define an infinitesimal unit ε with $\varepsilon < r$ for all $r \in \mathbb{R}$. Additionally, we make use of the standard part function $\mathrm{st}(\cdot) : {}^*\mathbb{R} \rightarrow \mathbb{R}$ which maps any $\varepsilon \in {}^*\mathbb{R}$ to its nearest real-valued representation.

Within our SCM we define an auxiliary variable $X_{\#\mathbf{I}} \in \{0..N\}$ that counts the number of active interventions. Upon intervening on the SCM we set $X_{\#\mathbf{I}} := |\,\mathbf{I}\,|$ with corresponding probability $\mathrm{P}(X_{\#\mathbf{I}} = n) = \varepsilon^n$. The probability assignment forms a valid distribution, as for $\mathrm{P}(X_{\#\mathbf{I}} = 0) = \varepsilon^0 = 1$ and for any $n \neq 0, \mathrm{P}(X_{\#\mathbf{I}}) = \mathrm{st}(\varepsilon^n) = 0$. The terms are additive and normalized $(\sum_{n=0}^{N} \varepsilon^n = \varepsilon^0 + \sum_{n=1}^{N} \varepsilon^n = 1 + \sum_{n=1}^{N} 0 = 1)$. In essence, we introduce an auxiliary variable within our model that gets added to the joint distribution of our SCM:

$$\mathrm{P}^{\mathbf{I}}_{\mathcal{M}}(\mathbf{X}) = \left[\prod\nolimits_{X_i \in \{\mathbf{X} \setminus \mathbf{I}\}} \mathrm{P}(X_i |\, \mathrm{pa}(X_i))\right] \cdot \mathrm{P}(X_{\#\mathbf{I}}) \tag{2}$$

With increasing numbers of interventions $\mathrm{P}(X_{\#\mathbf{I}})$ takes probabilities of higher-order infinitesimal values $\varepsilon^1, \varepsilon^2, \ldots$ which are totally ordered, irregardless of the remaining $\mathrm{P}_{\mathcal{M}}(\mathbf{X} \setminus X_{\#\mathbf{I}})$[3]. Taking the standard part of these probabilities $\mathrm{st}(\mathrm{P}^{\mathbf{I}}_{\mathcal{M}}(\mathbf{X})) = 0$ results in zero probability which underlines our intuition of considering interventions as external entities within the natural unfolding of our system. Importantly, in a scenario with no intervention present $\mathrm{P}(X_{\#\mathbf{I}})$ evaluates to $\varepsilon^0 = 1$, thus preserving probabilities in the unintervened case.

4 Discussion

In this paper, we discussed the use of backtracking counterfactuals as well as classical interventional counterfactuals for deriving explanations. In the light of obtaining arguments from structural causal models, we proposed to choose the most 'natural' explanation. That is, choosing explanations requiring the least number of interventions. Backtracking counterfactuals seem to be the more natural choice for supporting arguments as we do inherently try to avoid explaining counterfactual outcomes via arbitrary external interventions. On the other side, interventional counterfactuals can explain cases where backtracking reaches its limits. For this reason, we proposed a basic algorithm for gradually backing up from the interventionless setting towards explanations that involve more and

[3] Assuming the remaining terms only take values in $\mathbb{R}$.

more changes on the graph. Through this process we choose the explanation that requires the least number of interventions, therefore 'identifying the type of counterfactual at hand'. Eventually, we made a first attempt of inducing our preorder not only on the algorithmic level but also to encode it into the probabilities derived from the SCM itself by utilizing infinitesimal quantities.

Limitations. Our preference order currently only considers the number of interventions needed to explain the observed evidence. While the number of interventions might act as a sensible proxy for measuring the 'reasonability' of applied interventions, we expect that extended investigations on the impact and/or plausibility of different interventions should be done in the future.

Acknowledgements. The authors acknowledge the support of the German Science Foundation (DFG) project "Causality, Argumentation, and Machine Learning" (CAML2, KE 1686/3-2) of the SPP 1999 "Robust Argumentation Machines" (RATIO). It benefited from the Hessian Ministry of Higher Education, Research, Science and the Arts (HMWK; projects "The Third Wave of AI").

References

Aldrich, J.: Correlations genuine and spurious in Pearson and Yule. Stat. Sci. 364–376 (1995)

Baroni, P., Caminada, M., Giacomin, M.: An introduction to argumentation semantics. Knowl. Eng. Rev. **26**(4), 365–410 (2011)

Beckers, S., Halpern, J.Y.: Abstracting causal models. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 2678–2685 (2019)

Bex, F., Renooij, S., et al.: From arguments to constraints on a Bayesian network. In: COMMA, pp. 95–106 (2016)

Bochman, A.: Default logic as a species of causal reasoning. In: Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning, vol. 19, pp. 117–126 (2023)

Evans, M.G.: Causality and explanation in the logic of Aristotle. Philos. Phenomenol. Res. **19**(4), 466–485 (1959)

Falcon, A.: Aristotle on causality (2006)

Hume, D.: A Treatise of Human Nature. Clarendon Press, Oxford (1896)

Mackie, J.L.: The Cement of the Universe: A Study of Causation. Clarendon Press, Oxford (1980)

Pearl, J.: Embracing causality in default reasoning. Artif. Intell. **35**(2), 259–271 (1988)

Pearl, J.: Causality. Cambridge University Press, Cambridge (2009)

Robinson, A.: Non-Standard Analysis. Princeton University Press, Princeton (2016)

Rubenstein, P.K., et al.: Causal consistency of structural equation models. In: Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence (UAI 2017). AUAI Press (2017)

Timmer, S.T., Meyer, J.J.C., Prakken, H., Renooij, S., Verheij, B.: A two-phase method for extracting explanatory arguments from Bayesian networks. Int. J. Approx. Reason. **80**, 475–494 (2017)

Von Kügelgen, J., Mohamed, A., Beckers, S.: Backtracking counterfactuals. In: Conference on Causal Learning and Reasoning, pp. 177–196, PMLR (2023)

Wieten, R., Bex, F., Prakken, H., Renooij, S.: Constructing Bayesian network graphs from labeled arguments. In: Kern-Isberner, G., Ognjanović, Z. (eds.) ECSQARU 2019. LNCS (LNAI), vol. 11726, pp. 99–110. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-29765-7_9

Willig, M., Zečević, M., Dhami, D.S., Kersting, K.: Do not marginalize mechanisms, rather consolidate! In: Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS) (2023)

Interactive Argumentation, Recommendation and Personalization

BEA: Building Engaging Argumentation

Annalena Aicher[1,2](✉), Klaus Weber[2], Elisabeth André[2], Wolfgang Minker[1], and Stefan Ultes[3]

[1] Ulm University, Ulm, Germany
{annalena.aicher,wolfgang.minker}@uni-ulm.de
[2] University of Augsburg, Augsburg, Germany
{klaus.weber,elisabeth.andre,annalena.aicher}@uni-a.de
[3] University of Bamberg, Bamberg, Germany
stefan.ultes@uni-bamberg.de

Abstract. Exchanging arguments and knowledge in conversations is an intuitive way for humans to form opinions and reconcile opposing viewpoints. The vast amount of information available on the internet, often accessed through search engines, presents a considerable challenge. Managing and filtering this overwhelming wealth of data raises the potential for intellectual isolation. This can stem either from personalized searches that create "filter bubbles" by considering a user's history and preferences, or from the intrinsic, albeit unconscious, tendency of users to seek information that aligns with their existing beliefs, forming "self-imposed filter bubbles".

To address this issue, we introduce a model aimed at engaging the user in a critical examination of presented arguments and propose the use of a virtual agent engaging in a deliberative dialogue with human users to facilitate a fair and unbiased opinion formation. Our experiments have demonstrated the success of these models and their implementation. As a result, this work offers valuable insights for the design of future cooperative argumentative dialogue systems.

Keywords: Cooperative Argumentative Dialogue Systems · Reflective (User) Engagement (RUE) · Conversational Engagement · User Attention · User Focus · Gaze Tracking · Virtual Avatar

1 Introduction

Humans naturally form opinions and resolve differing perspectives through conversation, exchanging arguments and knowledge. Today's digital landscape offers

This work has been funded by the Deutsche Forschungsgemeinschaft within the project "BEA - Building Engaging Argumentation", Grant No. 455911629, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999).

A. Aiche and K. Weber—These authors contributed equally to this work and share first authorship. They are listed in alphabetical order.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 279–295, 2024.
https://doi.org/10.1007/978-3-031-63536-6_17

a wealth of opinions and information available online anytime. However, navigating and evaluating this abundance of sources can be a challenging task. Filter algorithms attempt to alleviate this challenge by personalizing content based on users' past requests, potentially leading to the creation of so-called "filter bubbles" [23] where users are exposed to information that is mainly consistent with their existing viewpoints. In addition to these external influences, an intrinsically motivated counterpart comes to the forefront. Even when having access to non-filtered sources, people tend to prioritize a biased subset of sources that echo or reinforce their pre-existing or convenient opinions, forming so-called" **self-imposed** filter bubbles" (SFB) [4,7,10]. To counteract this unintentional intellectual isolation, we aim to 1) engage the user in an intuitive, fair and unbiased process of opinion formation, 2) enable the user to explore a wide range of information naturally and intuitively and thus, to "build(ing) engaging argumentation". To this end, we introduce the cooperative argumentative dialogue system BEA embodied by a virtual agent participating in a deliberative dialogue with a human user. Unlike persuasive systems with competitive agendas, our system aims to offer a diverse and representative overview within the context of a conversation with the user.

The primary goal of BEA is to establish an interactive platform that motivates users to explore diverse perspectives and critically scrutinize information on diverse topics. To overcome the limitations of a one-sided conversation, BEA leverages a flexible natural language understanding and multiple in- and output modalities. To provide a basis for a thorough, well-rounded discussion, we derive the necessary specific characteristics of the argumentative dialogue. These critical features include, first, the user's demonstration of critical thinking and open-mindedness during the interaction with the agent, the so-called *reflective engagement*. And second, the user's motivation in sustaining the conversation with the system represented by a human-like avatar, the so-called *conversational engagement*. The following paper aims to elucidate the architecture of the argumentative dialogue system BEA and its associated modules. In particular, it explains the underlying model for the user's reflective engagement and the corresponding intervention strategy of BEA based on this model. As an evaluation of our models and implementation in BEA, we present the results of two studies, demonstrating 1) increased user attention and focus on relevant parts of the arguments due to BEA's intervention and 2) the positive impact of a human-like virtual avatar embodying BEA on conversational user engagement, trust, and the general perception of the system.

The remainder of this paper is as follows: Sect. 2 provides a short overview of relevant related work. Section 3 gives an overview of the different components of BEA, such as the formal argument structure, dialogue framework, interface etc. In Sect. 4, we present an approach to model the user's reflective engagement. In Sect. 5 BEA's contribution in enhancing the reflective and conversational user engagement is evaluated. Respective limitations of our work are discussed in Sect. 6, followed by a conclusion and outlook on future work in Sect. 7.

2 Related Work

In the following, we give a short overview of the related work on 1) argumentative dialogue systems, 2) reflective engagement and 3) conversational user engagement and virtual Avatars.

2.1 Argumentative Dialog Systems

Argumentative dialogue systems (ADS), conversational agents (CA), and Chatbots aim to interact with users through natural language by exchanging arguments. Most approaches to human-machine argumentation are embedded in a competitive setting [27,28]. They utilize different models to structure the interaction (similarity model to retrieve counterarguments [25], retrieval- and generative-based models [16]). In contrast, [5] introduced a cooperative argumentative dialogue system that provides arguments upon users' request without trying to persuade or win a debate against the user. We adopt this cooperative approach, as a mere confrontation with opposing arguments leads to cognitive dissonance [13], which can have a negative effect (defensive attitude [12]). Therefore, a confrontation in a competitive scenario is more likely to lead to rejection.

2.2 Reflective Engagement

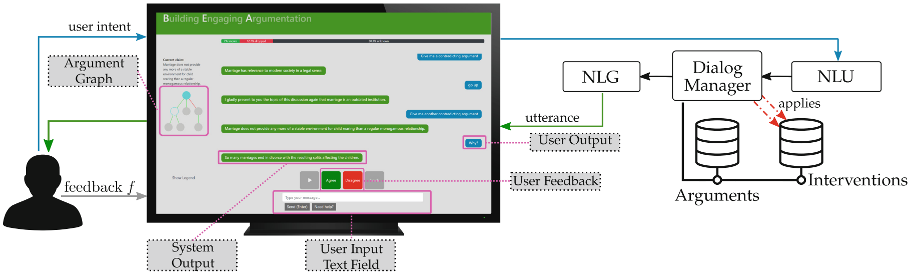
Reflective engagement (RE) in literature often denotes learners' active involvement in critically assessing their problem inquiry. Farr et al. [11] investigated markers of reflection in online discussions. Lyons et al. [17] emphasized the deliberate interruption of teaching practices for systematic questioning, highlighting the need for conscious awareness and adaptability. While existing research primarily explores RE in teaching-learning processes [14], our focus is on diverging viewpoints in argumentative scenarios. In contrast to methods like marker identification [11] our approach [30] integrates the user's stance and explored argumentation polarity for calculating reflective engagement, in line with [18]. Aicher et al. [3] introduced the reflective user engagement (RUE) score which aims for a balanced argument exploration of both sides. We extend this by rewarding users scrutinizing views opposing their current opinion. In their recent work [4,7] Aicher et al. introduced a model to determine the self-imposed filter bubble of the user and showed the effectiveness of the respective intervening "breaking strategy" to overcome the user's SFB [2].

2.3 Conversational User Engagement and Virtual Avatars

Current research often delves into the impact of self-identification with avatars in virtual spaces [20,26]). In contrast to virtual self-representations, our focus is on the influence of a virtual avatar as a discussion counterpart. Research in computer-mediated communication emphasizes that higher aesthetic and behavioral realism in avatars enhances user engagement, acceptance, and a sense of "social copresence." The results of Aseeri et al. [8] suggest that visual and non-verbal cues from different avatar representations affect user experience in cooperative tasks. In the context of argumentative dialogue systems, literature is very limited. Blount et al. [9] present an approach for participants to shape their own avatar appearance and how this impacts the course of an argumentative debate in the virtual sphere. However, there is a gap in analyzing the change in engagement, motivation, and perception when employing a virtual human-like avatar in a cooperative argumentative dialogue system instead of a chat-based interface, which we aim to address.

3 Prototype and Architecture of BEA

In the following, we give a short overview of prototype of BEA and its architecture, which is originally based on [5] and its extension [30].



Building Engaging Argumentation

7% known 12.7% dropped 80.3% unknown

Current claim: Marriage does not provide any more of a stable environment for child rearing than a regular monogamous relationship.

Give me a contradicting argument

Marriage has relevance to modern society in a legal sense.

go up

I gladly present to you the topic of this discussion again that marriage is an outdated institution.

Give me another contradicting argument

Marriage does not provide any more of a stable environment for child rearing than a regular monogamous relationship.

Why?

So many marriages end in divorce with the resulting splits affecting the children.

Show Legend

Type your message...

Fig. 1. Overview of system architecture in the interaction with a user.

3.1 System Architecture

Figure 1 sketches the architecture of our system. It consists of 1) an NLU (Natural Language Understanding), 2) a knowledge base of arguments, 3) a dialogue manager, 4) an NLG (Natural Language Generation), and 5) an intervention strategy.

Natural Language Understanding
The system uses an integrated natural language understanding framework (NLU) [1] to map the user's input to the available speech acts. The user can freely type their requests using a chat-input field to allow for a natural conversation. The NLU uses an intent classifier model consisting of two main components: a BERT Transformer Encoder and a bidirectional LSTM classifier.

Knowledge Base
The arguments that are available to the system throughout the interaction are encoded in an argument tree structure based on the argument annotation

scheme introduced in [29]. The knowledge base consists of a set of components L_t. It includes three types of argument components (*Major Claim*, *Claim*, and *Premise*) and two different directed relations (*support* and *attack*) between them. Within the scope of this work, relations are allowed from *Claims* to the *Major Claim*, *Premises* to *Claims* and *Premises* to *Premises*. If a component $\varphi_i \in L_t$ has a relation towards a component $\varphi_j \in L_t$, we say that φ_j is the target (of φ_i) and each component (apart from the Major Claim φ_0) has exactly one target. Hence, the arguments $\varPhi_i \in \boldsymbol{Args}$ that can be generated from such a structure have the form $\varPhi_i = (\varphi_i \Rightarrow \varphi_j)$ ($\widehat{=}$ *support*) or $\varPhi_i = (\varphi_i \Rightarrow \neg\varphi_j)$ ($\widehat{=}$ *attack*). Since each relation is unique and the difference between the three types of components is characterized solely by the allowed relations, each resulting structure can be represented as acyclic directed graph with argument components as nodes and relations as edges.

Throughout this work, we use the *idebate* dataset *Marriage is an outdated institution*[1] consisting of 72 arguments following the presented structure. The root argument is defined as $\varPhi_0 := \varphi_0$. Every argument $\varPhi_i \in Args$ has a stance $\in \{+, -\}$ towards φ_0 defined by the component relation and the respective position in the argument graph.

Table 1. Communication language L_c of the herein implemented dialogue system consisting of nine speech acts.

Speech Act	Description
System moves	
$argue(\varphi_i \Rightarrow \varphi_j)$	Present argument $\varphi_i \Rightarrow \varphi_j$
$jump\_to(\varphi_i)$	Jump to argument $\varPhi_i = \varphi_i \Rightarrow *$[a]
intervene	Suggest a challenger argument
User moves	
$why_{pro}(\varphi_i)$	Ask for a supporting component
$why_{con}(\varphi_i)$	Ask for an attacking component
$level_{up}$	Move level up
$agree(\varphi_i)$	Feedback to agree with a statement φ_i
$disagree(\varphi_i)$	Feedback to disagree with a statement φ_i
$confirm/reject$	Confirm/Reject intervention

[a] $* \in \{\varphi, \neg\varphi\}$

Dialogue Manager

The dialogue manager has access to different knowledge bases and provides a communication language L_c, which includes the speech acts available to the

[1] https://idebate.net/resources/debatabase.

user and system (see Table 1). These speech acts are tailored to suit the purpose of a specific dialogue system and can be modified accordingly. It manages the dialogue between the system and the user and ensures logical consistency. Furthermore, the dialogue manager stores the current dialogue state, i.e., the complete dialogue history, which arguments have been presented, the current position within the argument tree, and allowed speech acts. For instance, if an argument Φ_i is a leaf node, $why_{pro}(\varphi_i)$ is not allowed. If the user requests a new argument (why_x), the system selects a random argument from all arguments fitting the requested relation $x \in \{pro, con\}$.

Natural Language Generation

The system generates a textual response, wherein the Natural Language Generation (NLG) relies on the original surface text of the argument components, denoted as $\varphi_i \in L_t$. These annotated sentences were manually adjusted in terms of grammatical syntax to create independent utterances, serving as templates for the corresponding system responses. In order to add some diversity, a collection of natural language formulations was created for each speech act. During the response generation, the explicit formulation is chosen from this list randomly. To structure the dialogue as comprehensible and understandable for the user as possible, and to clearly present contextual connections, the system employs transitional phrases such as *Let us return to the previous argument, that ...* or *This claim is supported by the argument that ...*. Please note that the system presents all arguments in a neutral manner, without taking a stance of its own.

Intervention Strategy

The intervention keeps track of the user's reflective engagement (RUE) (RUE, see Sect. 4 for calculations) and intervenes if necessary, i.e., it suggests considering an opposing viewpoint and presents a challenger argument if the user agrees. Let $l^+, l^- \in L_t$ be the set of all valid[2] argument components that the system can present. For every user request, the system simulates the RUE_{Φ_0} for all valid challenger arguments and returns the component that maximizes it. The intervention takes place if and only if the simulated RUE is greater than the maximum possible RUE for all available non-challenger arguments, i.e., if the user requests a supporting argument, the system checks if

$$\max_{\varphi_i \in l^-}(sim\_rue(\varphi_i)) > \max_{\varphi_i \in l^+}(sim\_rue(\varphi_i)) \tag{1}$$

and, if so, returns $\varphi_i = \arg\max_{\varphi_i \in l^-}(sim\_rue(\varphi_i))$, and suggests it to the user, who can accept or reject it. In case of rejection, the system proceeds with the initial user request.

3.2 User Interface

Table 2 sketches a sample dialogue between the user and system.

[2] A valid pro (+) and con (-) argument component φ_i is an unheard component that links to any component φ_j that has already been presented to the user.

Table 2. Exemplary dialogue between the system and the user about the topic *"Marriage is an outdated institution"* with intervention condition.

Actor	Utterance/Action	Speech Acts
system	*Let's discuss that marriage is an outdated institution.*	claim(φ_0)
User	Give me an argument in favor.	$why_{pro}(\varphi_0)$
system	*Marriage does not provide a more stable environment for child rearing than a regular monogamous relationship.*	$argue(\varphi_1 \Rightarrow \ \varphi_0)$
User	*Clicks green button "Agree" on interface.*	$agree(\varphi_1)$
system	*Alright, I noted your feedback. Let us return to the topic of this discussion.*	-
User	Could you elaborate this further?	$why_{pro}(\varphi_0)$
system	*I think we should look at the opposite point of view. Alright?*	*intervene*
User	Yes	$confirm$
system	*Marriage removes the transient and casual aspects of a monogamous relationship, thus giving a child a far more stable environment.*	$argue(\varphi_2 \Rightarrow \ \neg\varphi_0)$

The user interface consists of four components: 1) the dialogue history showing the system output and user input, 2) the user input text field, 3) the user feedback buttons, and 4) the graphically displayed information about the argument structure. Furthermore, instead of the dialogue history in the middle of the screen, a human-like 3D avatar can be displayed in the same position, presenting the arguments in spoken language (see Fig. 4). The dialogue history is depicted on the right side of the screen in this configuration.

A classic chat design displays the system's *textual* response. The interface provides a *text* (chat) input where users can formulate their requests. To allow for feedback on whether users agree with the current argument or not, there are two buttons (*Agree* and *Disagree*) that the user can use at any time during interaction. Without feedback, the system considers a neutral user stance. On the left side the browser window left side, the current argument graph displays a visual representation of all arguments of the respective *Root Claim*[3]. An outlined turquoise node denotes the user's current position, the already discussed arguments are shown in solid turquoise, and unheard arguments are in grey. The edges between the nodes show a supporting relation in green and an attacking one in red. Users can choose whether to ask for a pro or con argument or how they want to navigate through the argument tree.

[3] We define a *Root Claim* as a *Claim* φ_j which directly *attacks* or *supports* the *Major Claim* φ_0.

4 Modeling Reflective Engagement

In the following, we give a short overview of the reflective engagement model (for details see [3,30]).

A set of arguments with the same target argument $\varPhi_i$ is denoted as $P_{\varPhi_i}$. If it is in favor of stance +, it is denoted as $P^+_{\varPhi_i}$ and $P^-_{\varPhi_i}$ elsewise. The set of all visited arguments $\varPhi_j$ with target argument $\varPhi_i$ is denoted as $P_{\varPhi_i,v}$.

We define the user's focus for argument $\varPhi_i$ based on visited pro and con arguments as:

$$focus_{\varPhi_i} = \frac{\left|P^+_{\varPhi_i,v}\right| - \left|P^-_{\varPhi_i,v}\right|}{|P_{\varPhi_i,v}|} \in [-1,1]. \tag{2}$$

It is easy to verify that the more arguments of a certain stance are selected by the user, the more the focus shifts in the direction of the respective stance.

The overall normalized user focus $\mathcal{F} \in [-1,1]$ is then defined by summing up and normalizing the $focus_{\varPhi_i}$:

$$\mathcal{F} := \frac{\sum_{\varPhi_k \in Args} focus_{\varPhi_k}}{|Args|} \tag{3}$$

During the interaction with the system, users can give feedback on whether or not they agree or disagree with any argument $\varPhi_i \in Args$. Considering the hierarchical structure of arguments, the system uses this feedback to compute the user's stance $e_{\varPhi_j} \in [0,1]$ of any argument $\varPhi_j \in Args$ considering the feedback for this respective argument and the feedback for all arguments in the subtree with $\varPhi_j$ as root (following the approach of [31]).

The user's reflective engagement considers the weighted user's focus by making use of the inverted correlation of stance and focus (Eq. 4). This is based on the assumption that users with a particular stance are likely to focus more on arguments that are in line with their stance. Users with a higher level of RE tend to look at claims that support an opposite view as well [24].

$$RUE = 1 - \left| e_{\varPhi_0} - \left(1 - \frac{\mathcal{F}+1}{2}\right)\right| \tag{4}$$

After inverting the normalized focus, the difference between user stance and focus is taken to compute RUE, i.e., the more the focus aligns with the user's stance, the lower the RUE and vice versa. This approach ensures that if the user stance is positive (+), the system intervenes to suggest the user choose more con arguments (challenger arguments of pro arguments) and vice versa.

5 Evaluation

In the following we give an overview of the results of to our two previously published evaluation studies[4], aimed to analyze the influence of 1) the intervention

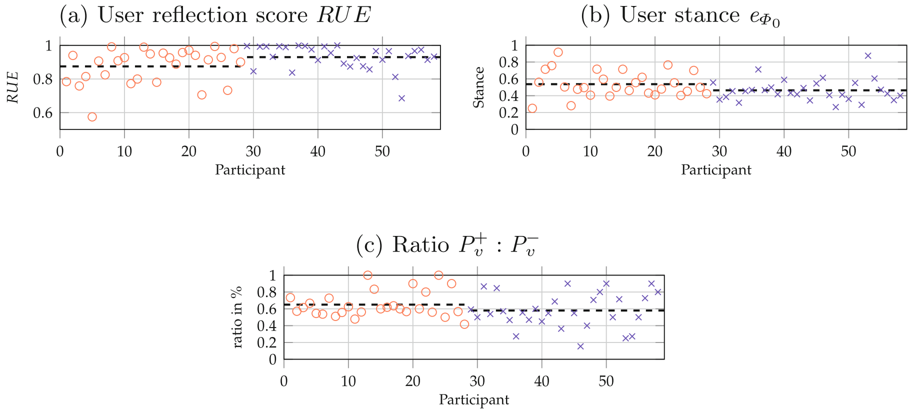
[4] Due to space limitations, only essential results are presented here; further details are available in our cited work.

on the user's focus on challenger arguments [30] and 2) a virtual human-like avatar on the general user perception and conversational user engagement [6].

5.1 Study 1 [30]: Analyzing Focus on Challenger Arguments

The first study was conducted online via the crowdsourcing platform "Crowdee[5]" with 58 participants (aged 18–63) divided into two groups (an experimental group with intervention and a control group without intervention) from the UK, US, and Australia (English native speakers to avoid language barrier effects). The study setup used the chat-based output modality. After an introduction to the system (short text and description of how to interact with the system), the users were advised to explore enough arguments to build a well-founded opinion on the topic *Marriage is an outdated institution*. The participants were not told anything about the underlying reflection model but only to select at least ten arguments. In addition, they were asked to rate their opinion on the topic on a 5-point Likert scale, which normalized in $[0, 1]$ displayed the initial user stance e_0. During the study, we collected the following data anonymously:

1. Measured user reflection score RUE (Fig. 2a).
2. User stance e_{Φ_0} (Fig. 2b)
3. Set of visited arguments P_v^+ and P_v^- (Fig. 2c).


(a) User reflection score RUE

(b) User stance e_{Φ_0}

(c) Ratio $P_v^+ : P_v^-$

Fig. 2. Collected data: Reflection score RUE, user stance e_{Φ_0}, and focus $\mathcal{F}$ of both the experimental group (x) and the control group (o).

Statistical Analysis: Concerning the calculated RUE score (Fig. 2a), the homogeneity of variances was falsified utilizing the Levene's test ($F = 5.64$, $p = .021$) and the assumption of a normal distribution using the Shapiro-Wilk

[5] https://www.crowdee.com/.

test ($W = 0.895$, $p < .001, W = 0.874$). Thus, we applied the *Mann-Whitney-U test* showing a main effect of intervention on RUE ($U = 273$, $n_1 = 30$, $n_2 = 28$, $p \leq .01$). In addition to that, we also checked the total amount of interventions. There were 262 interventions in the experimental condition (8.73 per user), 201 of which were accepted by the user, which is an acceptance rate of 76%.

To investigate the main effect of intervention on challenger arguments, we analyzed how many participants were more engaged with *challenger arguments* by comparing the amount of pro and con arguments that the user heard to the user stance, e.g., if the user stance is negative ($e_{\Phi_0} < 0.5$) and more pro than con arguments were heard ($P_v^+ > P_v^-$), it implicates a strong challenged engagement (see Table 3).

Table 3. Contingency table of focus on challenger arguments per condition.

Condition	Challenger arg.	Non-challenger arg.	Total
Experimental	24	6	30
Control	15	13	28
Marginal Column Total	39	19	58

We found that with intervention, nearly 80% of participants were more engaged with *challenger arguments*, while only 53% in the control condition did so, which is a total increase of 51%. A chi-square test of independence [19] was performed to examine the relation between *condition* and *engagement with challenger arguments* showing a significant relation ($X^2(1, N = 58) = 4.5924, p = .032$).

Analyzing the main effect of intervention on the total percentage of heard *challenger arguments* revealed a large significant main effect (*T-Test*, $t(56) = 2.0903, p = .02, d = .55$, Fig. 3), proving that the users in the experimental group focused significantly more on *challenger arguments* than in the control condition.

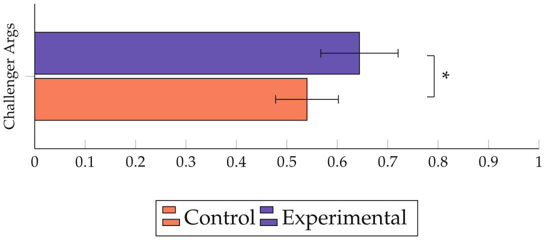
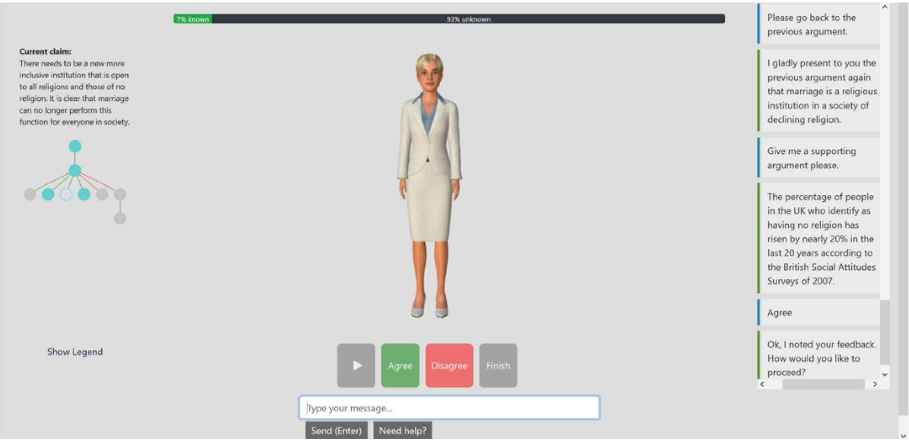


Fig. 3. Means including 95% confidence interval denoted by bars of focus on challenger arguments. (*) $p < .05$.

5.2 Study 2 [6]: Influence of Avatar Interface



Current claim:
There needs to be a new more inclusive institution that is open to all religions and those of no religion. It is clear that marriage can no longer perform this function for everyone in society.

Please go back to the previous argument.

I gladly present to you the previous argument again that marriage is a religious institution in a society of declining religion.

Give me a supporting argument please.

The percentage of people in the UK who identify as having no religion has risen by nearly 20% in the last 20 years according to the British Social Attitudes Surveys of 2007.

Agree

Ok, I noted your feedback. How would you like to proceed?

Fig. 4. User interface with avatar. Above the chat-input line four buttons and the virtual avatar are shown. The dialogue history is placed on the right side of the screen.

After having shown the positive impact of BEA's intervention on the user's focus and reflective engagement, this subsection focuses on analysing how to create a enjoyable, natural discussion and maintain the user motivation to interact with BEA. Therefore, we examine the impact of avatar versus non-avatar interfaces on user perception, engagement, and trust in argumentative dialogue systems, we conducted a crowdsourcing study. Eighty-four participants (aged 18–65; 52 female, 31 male, 1 "other/do not want to say") were divided into two groups: 46 interacted with a virtual avatar interface (avatar system) illustrated in Fig. 4, and 38 with a non-avatar interface (non-avatar system) (similar to the screen in Fig. 1). Both systems were identical, differing only in the graphical user interface (chat-based output for the non-avatar system, spoken avatar output for the avatar system). The avatar interface utilized the Charamel$^{\text{TM}}$ avatar[6] with synthetic speech utilizing Nuance TTS and Amazon Polly Voices[7].

Participants interacted significantly longer with the avatar, influenced by the avatar's spoken utterance and response delays caused by the avatar server. The participants rated statements on a 5-point Likert scale (1="Totally disagree", 5="Totally agree") across three questionnaires. In addition to the three questionnaires, we asked the participants to rate their opinion and interest in the topic "Marriage is an outdated institution" before ("pre") and after ("post") the interaction. There is no significant difference between the two participant groups (*Mann-Whitney-U-Test*) or between the pre- and post-conditions (*Wilcoxon*

[6] https://www.charamel.com/competence/avatare.

[7] https://docs.aws.amazon.com/polly/latest/dg/voicelist.html.

signed-rank test) regarding user opinion. This is important to avoid the risk that the avatar itself biases and manipulates user opinion. While the differences in user interest are not yet significant, they are still noticeable for the avatar group ($Mean_{pre} = 3.56$, $Mean_{post} = 3.81$,$p = .074$, $r = .195$) compared to the non-avatar one ($Mean_{pre} = 3.58$, $Mean_{post} = 3.53$,$p = .614$, $r = .055$). This implies a slight tendency that the user interest is positively influenced by the avatar.

First questionnaire (adopted from a questionnaire according to ITU-T Recommendation P.851) [22][8]: consists of 39 single items and measures the user's general impression of the system. Its items are grouped by the following aspects: information provided by the system (IPS), communication with the system (COM), system behavior (SB), dialogue (DI), user's impression of the system (UIS), acceptability (ACC), and argumentation (ARG)[9].

Even though the differences between the avatar and non-avatar group regarding the merged aspects IPS, COM, SB, DI, UIS, ACC and ARG are insignificant (*Mann-Whitney-U-Test*), we can perceive some consistent, aspect-overlapping tendency. Regarding the aspects IPS, COM, UIS and ACC neither the single item analysis nor the merged analysis showed any significant differences. Likewise also for the three other aspects (SB, DI and ARG) the merged analysis did not show any significant differences, but we could still perceive some consistent, aspect-overlapping tendency. Especially regarding the perceived naturalness and the engagement users felt, the avatar system is rated significantly better.

Second questionnaire: consists of 12 items [21] and measures the conversational engagement. Its items are grouped by the following aspects: Focused attention (FA), perceived usability (PU), aesthetic appeal (AE) and Reward (RW). The findings revealed the avatar system's engaging effect, as indicated by better ratings across all items and especially significant for the impression that using the system was worthwhile ($p_{RW1} = .022, r_{RW1} = 0.25$). However, perceived usability needs improvement, particularly regarding automated speech recognition errors and the explanation of the system's reaction if the user was not understood correctly.

Together with the voluntary the significant difference in the expected help the system should have provided, this implies that the avatar on one hand side tends to raise the expectation to that of a human conversational partner. Thus, fulfilling these expectations could lead to a significantly stronger acceptance comparable to a human conversational partner.

Third questionnaire [15][10]**:** consists of 11 items and measures user trust. Its items are grouped by the following aspects: understanding/predictability (UP), familiarity (F), propensity to trust (PT) and trust in automation (TA). The users tend to trust the avatar system more than the non-avatar one, especially regarding their propensity to trust ($p_{PT} = 0.015$, $r_{PT} = 0.266$). Both participant

[8] Such questionnaires can be used to evaluate the quality of speech-based services.

[9] Self-added aspect since this is not captured by standardized questionnaires.

[10] This questionnaire was developed to measure trust in automation.

groups do not differ noticeably in the familiarity with similar systems, which implies an inclination towards trusting the avatar system more. Thus, especially by individualizing the avatar further, we believe to increase the user trust and support a well-founded opinion building.

In summary, our findings support using an avatar interface in ADS, emphasizing its potential to enhance user engagement and trust without manipulating their opinion. Addressing usability concerns can further optimize the user satisfaction and it will become easier to maintain the interaction.

6 Limitations

This paper, however, is subject to some limitations that will be addressed in future research. First, in our second study, we did not compare different avatar settings personalized to individual users, nor did we conduct per-participant analyses. For this exploratory study, we opted for an easily implementable, widely accessible, representative avatar, rather than one that is highly individualized. The comparison to a purely chat-based interface aimed to evaluate the influence of avatars on argumentative interactions in general. The focus was to determine whether the mere visualization of an avatar leads to a bias in opinion formation or influences the perception of the provided argumentative content and conversational engagement of users even though the avatar has not been personalized. However, future research in this domain should investigate the impact of personalized avatar features and how individual participants perceive the interaction, the provided argument content, and the avatar's personality traits (e.g., dominance, friendliness/pleasure). In order to capture the full range of experiences and perspectives of the participants in future studies, it may be beneficial to supplement the study with qualitative data in the form of participant interviews (e.g. by free text responses). Furthermore, it needs to be mentioned, that due to limited space we did not discuss user comments in detail. Moreover in future work we will put a specific focus on analyzing aspects directly related to argumentation, such as the perception of argument strategy/selection (regarding consistency, quality, persuasiveness, etc.) in relation to differences in avatar modeling.

Another limitation is that both user studies focus solely on one topic ("Marriage is an outdated institution") derived from a single source. We selected this topic because its dataset fulfills our criteria of being sufficiently large, balanced in terms of argument stance (pro/con), of high quality, and having depth in arguments. Although it appears suitable for a proof-of-principle study, the scalability of our findings needs to be demonstrated concerning other topics.

Moreover, we emphasize that while the user-agent interaction may seem constrained and artificial because users are unable to introduce counterarguments, this decision was deliberate. The aim of the argumentative dialogue system is to neutrally confront users with pro/con arguments on a given topic, allowing them to explore without being directly engaged in a persuasive discussion. As pointed out by [24] due to the users' tendency to defend their own view, a system which confronts them with an opposing stance might not lead to critical reflection but rather the opposite. However, we aim for a more natural exchange in future work. We intend to explore how users can introduce their own arguments without exacerbating their self-imposed filter bubble. To achieve this, we propose investigating the approach of dynamically searching for relevant arguments in real-time, which can support both viewpoints and supplement the existing argumentative structure with arguments that have not yet been part of the argumentative discourse.

One final limitation to mention here is that the study results presented herein have predominantly been introduced separately in our earlier publications. The intention behind this is to integrate and merge findings from previously separately explored dimensions of reflective and conversational engagement within argumentative discourse involving real users. This synthesis shall provide a basis for addressing questions that involve both forms of engagement, conversational and reflective engagement, to enhance argumentative discussions with individual users in future work. Thus in this paper, we want to emphasize the importance of acknowledging the significance of investigating both conversational and reflective engagement to attain a critically-reflected and enjoyable interaction with real users within the context of robust argumentation machines.

7 Conclusion and Future Work

In this work we introduced an approach to build engaging argumentation. Therefore, we focused on exploring methods to enhance the reflective and conversational engagement of users in the interaction with the cooperative argumentative dialogue system BEA. To account for a critical reflection of arguments, we introduce a model that characterizes reflective user engagement (RUE) and propose a corresponding intervention strategy. To maintain the user motivation to continue the interaction, especially when guiding them to explore opposing viewpoints without explicit request, we suggest incorporating a virtual, human-like avatar to embody the system.

The first of two presented user studies demonstrates that BEA's intervention not only increased RUE but also enhanced the user focus on challenger arguments. Results from a second user study imply the positive impact of a human-like virtual avatar embodying BEA on conversational user engagement, trust, and overall system perception. This suggests that a human-like design generates expectations for communication and assistance, akin to interactions with a human conversational partner without manipulating the user's opinion. However, errors or delays in the avatar's response time can have a noticeable adverse impact, emphasizing the need for addressing these issues in future refinements of the avatar setting.

Also, in future work, we aim to explore how personalizing and individualizing avatars influence users' perception of argument content (persuasiveness, etc.) and user trust and motivation, particularly in relation to specific avatar traits. Through qualitative analyses, we hope to gain more insight into the factors that contribute to manipulating the rational perception of argument content when modeling virtual, argumentative avatars. And we will investigate the potential of personalized avatar implementation to enhance interaction in argumentative dialogue systems, through social presence influence, positive feedback, and emotional connection, with the intention to increased user engagement and satisfaction.

To ensure scalability of the described results and introduced approaches, we aim to test them on datasets covering other controversial topics from various sources. Additionally, we aim to enhance natural interaction by allowing users to contribute their own aspects/arguments while ensuring that the user's self-imposed filter bubble does not reinforce.

In conclusion, based on the established connection between reflective and conversational engagement in argumentative interaction with real users, we aim to further explore the interdependencies between them and their impacts. Therefore, this work presents important implications for designing a critically-reflected and enjoyable interaction within the context of robust argumentation machines.

References

1. Abro, W.A., Aicher, A., Rach, N., Ultes, S., Minker, W., Qi, G.: Natural language understanding for argumentative dialogue systems in the opinion building domain. Knowl. Based Syst. **242**, 108318 (2022). https://www.sciencedirect.com/science/article/pii/S0950705122001149
2. Aicher, A., Kornmueller, D., Matsuda, Y., Ultes, S., Minker, W., Yasumoto, K.: Towards breaking the self-imposed filter bubble in argumentative dialogues. In: Stoyanchev, S., Joty, S., Schlangen, D., Dusek, O., Kennington, C., Alikhani, M. (eds.) Proceedings of the 24th Meeting of the Special Interest Group on Discourse and Dialogue, pp. 593–604, September 2023. https://aclanthology.org/2023.sigdial-1.56
3. Aicher, A., Minker, W., Ultes, S.: Determination of Reflective User Engagement in Argumentative Dialogue Systems (2021)
4. Aicher, A., Minker, W., Ultes, S.: Towards modelling self-imposed filter bubbles in argumentative dialogue systems. In: Proceedings of the 13th LREC, pp. 4126–4134, June 2022. https://aclanthology.org/2022.lrec-1.438
5. Aicher, A., Rach, N., Minker, W., Ultes, S.: Opinion building based on the argumentative dialogue system BEA. In: Marchi, E., Siniscalchi, S.M., Cumani, S., Salerno, V.M., Li, H. (eds.) Increasing Naturalness and Flexibility in Spoken Dialogue Interaction. LNEE, vol. 714, pp. 307–318. Springer, Singapore (2021). https://doi.org/10.1007/978-981-15-9323-9_27
6. Aicher, A., Weber, K., André, E., Minker, W., Ultes, S.: The influence of avatar interfaces on argumentative dialogues. In: Proceedings of the 23rd ACM International Conference on Intelligent Virtual Agents. IVA 2023 (2023).https://doi.org/10.1145/3570945.3607343
7. Aicher, A.B., Kornmüller, D., Minker, W., Ultes, S.: Self-imposed filter bubble model for argumentative dialogues. In: Proceedings of the 5th International Conference on Conversational User Interfaces. CUI 2023 (2023).https://doi.org/10.1145/3571884.3597131

8. Aseeri, S., Interrante, V.: The influence of avatar representation on interpersonal communication in virtual social environments. IEEE Trans. Visual Comput. Graph. **27**(5), 2608–2617 (2021). https://doi.org/10.1109/TVCG.2021.3067783
9. Blount, T., Millard, D.E., Weal, M.J.: On the role of avatars in argumentation. In: Proceedings of the 2015 Workshop on Narrative & Hypertext, pp. 17–19. NHT 2015, Association for Computing Machinery (2015).https://doi.org/10.1145/2804565.2804569
10. Ekström, A.G., Niehorster, D.C., Olsson, E.J.: Self-imposed filter bubbles: selective attention and exposure in online search. Comput. Human Behav. Rep. **7**, 100226 (2022). https://doi.org/10.1016/j.chbr.2022.100226
11. Farr, F., Riordan, E.: Students' engagement in reflective tasks: an investigation of interactive and non-interactive discourse corpora. Classroom Disc. **3**(2), 129–146 (2012). https://doi.org/10.1080/19463014.2012.716622
12. Harmon-Jones, E.: Cognitive dissonance and experienced negative affect: evidence that dissonance increases experienced negative affect even in the absence of aversive consequences. Pers. Soc. Psychol. Bull. **26**(12), 1490–1501 (2000). https://doi.org/10.1177/01461672002612004
13. Hart, W., Albarracín, D., Eagly, A.H., Brechan, I., Lindberg, M.J., Merrill, L.: Feeling validated versus being correct: a meta-analysis of selective exposure to information. Psychol. Bull. **135**(4), 555 (2009)
14. Kong, S.C., Song, Y.: An experience of personalized learning hub initiative embedding BYOD for reflective engagement in higher education. Comput. Educ. **88**, 227–240 (2015). https://doi.org/10.1016/j.compedu.2015.06.003
15. Körber, M.: Theoretical considerations and development of a questionnaire to measure trust in automation. In: Bagnara, S., Tartaglia, R., Albolino, S., Alexander, T., Fujita, Y. (eds.) IEA 2018. AISC, vol. 823, pp. 13–30. Springer, Cham (2019). https://doi.org/10.1007/978-3-319-96074-6_2
16. Le, D.T., Nguyen, C.T., Nguyen, K.A.: Dave the debater: a retrieval-based and generative argumentative dialogue agent. In: Proceedings of the 5th Workshop on Argument Mining, pp. 121–130 (2018).https://doi.org/10.18653/v1/W18-5215
17. Lyons, N.: Reflective engagement as professional development in the lives of university teachers. Teach. Teach. **12**(2), 151–168 (2006)
18. Mason, M.: Critical thinking and learning. Educ. Philos. Theory **39**(4), 339–349 (2007). https://doi.org/10.1111/j.1469-5812.2007.00343.x
19. McHugh, M.L.: The chi-square test of independence. Biochemia medica **23**(2), 143–149 (2013).https://doi.org/10.11613/BM.2013.018
20. Mohd Tuah, N., Wanick, V., Ranchhod, A., Wills, G.: Exploring avatar roles for motivational effects in gameful environments. EAI Endorsed Trans. Creat. Tech. **4**, 153055 (2017).https://doi.org/10.4108/eai.4-9-2017.153055
21. O'Brien, H.L., Cairns, P., Hall, M.: A practical approach to measuring user engagement with the refined user engagement scale (UES) and new UES short form. Int. J. Human-Comput. Stud. **112**, 28–39 (2018)
22. P.851, I.T.R.: Subjective quality evaluation of telephone services based on spoken dialogue systems (11/2003). Int. Telecommunication Union, November 2003
23. Pariser, E.: The Filter Bubble: How the New Personalized Web Is Changing What We Read and How We Think. Penguin Books, USA (2012)
24. Paul, R.W.: Critical and reflective thinking: A philosophical perspective, pp. 445–494 (1990), north Central Regional USA
25. Rakshit, G., Bowden, K., Reed, L., Misra, A., Walker, M.: Debbie, the debate bot of the future. In: IWSDS 2017, pp. 45–52, June 2017

26. Ratan, R., Rikard, R., Wanek, C., McKinley, M., Johnson, L., Sah, Y.J.: Introducing avatarification: an experimental examination of how avatars influence student motivation (2016).https://doi.org/10.1109/HICSS.2016.15
27. Rosenfeld, A., Kraus, S.: Strategical argumentative agent for human persuasion. In: ECAI 2016, pp. 320–328 (2016).https://doi.org/10.3233/978-1-61499-672-9-320
28. Slonim, N., et al.: An autonomous debating system. Nature **591**(7850), 379–384 (2021). https://doi.org/10.1038/s41586-021-03215-w
29. Stab, C., Gurevych, I.: Annotating argument components and relations in persuasive essays. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics Technical Papers, pp. 1501–1510, August 2014. https://aclanthology.org/C14-1142
30. Weber, K., Aicher, A., Minker, W., Ultes, S., André, E.: Fostering user engagement in the critical reflection of arguments. In: Proceedings of the 13th International Workshop On Spoken Dialogue Systems (IWSDS) (2023), accepted for publication
31. Weber, K., et al.: Predicting persuasive effectiveness for multimodal behavior adaptation using bipolar weighted argument graphs, pp. 1476–1484 (2020)

Deciphering Personal Argument Styles – A Comprehensive Approach to Analyzing Linguistic Properties of Argument Preferences

Mark-Matthias Zymla(✉), Raphael Buchmüller, Miriam Butt, and Daniel Keim

University of Konstanz, Universitätsstraße 10, 78464 Konstanz, Germany
{Mark-Matthias.Zymla,Raphael.Buchmueller,Miriam.Butt,
DanielKeim}@uni-konstanz.de

Abstract. In this paper, we introduce an application for exploring the effect of linguistic features on personalized argument preferences. These individual preferences are derived by measuring the impact of linguistic features on pairwise comparisons between arguments. The insights derived from this are, in turn, useful for studies of argument quality. To conduct this research, we have developed a new pipeline that covers three major components: data collection, argument comparison labeling, and data exploration, incorporating linguistic annotations of arguments and preference data. The first component has resulted in a novel corpus consisting of minimal pairs of arguments: the comparable argument corpus. For the second component, we have developed a visual interactive labeling system that structures the annotation process of pairwise comparisons. Through these annotations, we extract patterns of argument preferences using Gaussian Process Preference Learning based on linguistic feature vectors. The corresponding, personalized models are used to identify relevant features to explain argument preferences. By training individual models for different users, we gain information that allows us to compare different user groups, identifying different argumentation preferences across groups. Each of these steps is supported by novel visual analytics dashboards, facilitating data collection and annotation steps and enabling the exploration of personal preferences.

Keywords: argument quality · argument preferences · visual analytics

This work has been funded by the **Deutsche Forschungsgemeinschaft (DFG)** within the project **CUEPAQ**, Grant Number 455910360, as part of the Priority Program **"Robust Argumentation Machines (RATIO)"** (SPP-1999). We are grateful to the participants of the Priority program RATIO for discussions at various project meetings. Furthermore, we would like to thank Mennatallah El-Assady, Annette Hautli-Janisz, Chris Reed, and Rita Sevastjanova for providing crucial feedback We are also very thankful to Fabian Sperrle for his contributions to the project.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 296–314, 2024.
https://doi.org/10.1007/978-3-031-63536-6_18

1 Introduction

This paper addresses a central question within research on argumentation, namely: What makes a good argument? [29,41,43–45]. The literature so far has established, that the quality of an argument has many dimensions, which pertain to the content of the arguments themselves as well as their rhetorical "packaging". In our project Visual Analytics and Linguistics for Capturing, Understanding, and Explaining Personalized Argument Quality (CUEPAQ), we have built on our expertise in the linguistic analysis of argumentation [11,14,38] to explore the hypothesis that argument preferences are, in fact, often more subjective than the current state of the art in the literature leads us to believe (cf. [41]). More concretely, the project focuses on the effect of linguistic features on personalized argument preferences.

For this, we have developed a new application, the CUEPipe. This pipeline allows researchers to generate data sets for assessing personalized argument preferences as well as annotating these data sets for argument preference. Expecting different results from different annotators, we also provide a platform for exploring personalized argument models learned from the annotations. Thus, the CUEPipe allows linguists to investigate argument preferences, including our claim that argument preferences are, to some extent, subjective. In this paper, we describe three major components of the application:

i. An interface for generating a corpus of arguments and exploring its linguistic feature diversity
ii. An interface for labeling pairwise comparisons between arguments
iii. An interface for exploring personal argument preferences

We illustrate each of these steps based on a proof-of-concept use case by reporting our own experiences with the application and the results of a user study tailored towards testing the visual interactive labeling aspect of the application and the exploration of personal argument preferences. For this, our declared goal was to explore whether how we attribute beliefs to different entities affects how we perceive the corresponding arguments. We do this by looking at how propositional attitude verbs affect argument preferences.

The paper is structured as follows: In the next section, we describe the concepts explored in CUEPAQ in more detail. In Sect. 3, we describe how these concepts relate to the CUEPipe and in Sect. 4, we describe our pilot use case involving user studies. Section 6 concludes.

2 Background

One main goal of our research is to investigate the impact of linguistic features on argument preferences in a controlled manner. To achieve this, we drastically simplify the complexity often attributed to the structure of arguments, as becomes apparent when investigating the topic of argumentation schemes (e.g., [23,31,46,47]). As such, we rely on the simple idea that "Argumentation is aimed at increasing or (decreasing) the acceptability of a controversial standpoint" [42, p. 4].[1] In the next section, we motivate this decision.

2.1 Argument Data

We treat arguments as tuples (*premise*,*conclusion*,*relation*), following basic (computational) argumentation schemes [30]. The *premise* and the *conclusion* are unmodified linguistic expressions.[2] They stand in a specified relation to the argument and are taken to be a member of the set {*support*,*attack*}. The *support* relation indicates that the premise increases the acceptability of the conclusion, while the *attack* relation aims at decreasing the acceptability of the conclusion. An example is illustrated in (1).

(1) Covid has a 2% mortality rate. $\rightarrow_{support}$ Covid is dangerous.

One of the reasons to focus on the simple arguments is to enable the contrastive study of linguistic features by means of minimal pairs. Minimal pairs originated in the linguistic study of sounds and are used to help determine distinctive classes, for example, to determine the phonemes of a language. Beyond phonology, the concept has been applied to different kinds of minimal pairs, prominently syntactic minimal pairs, which have been used, for example, in language acquisition research [10,20]. Similarly, minimal pair data has been used to judge the linguistic ability of machine and deep learning systems (see, e.g., [27,48]). Our goal is to investigate whether minimal changes affect the judgment of argument preferences. More concretely, our corpus helps to explore how minimal changes in the premise affect the acceptability of the conclusion of an argument. A typical minimal pair in our corpus is exemplified by (1) vs. (2). As can be seen, our minimal pairs are based on the choice of lexical items that make up an argument. The term *minimal* refers to the addition, removal, or change of at most one word.

(2) Covid only has a 2% mortality rate. $\rightarrow_{support}$ Covid is dangerous.

As discussed in Sects. 3 and 4, these minimal pairs help provide balanced corpora for research into individual preferences as tied to linguistic features.

2.2 Argument Preferences

According to recent work on the assessment of argument quality, argument preferences are affected by various dimensions [43,44]. However, these dimensions

[1] While we restricted ourselves to simple argument types, the methods presented in this paper are applicable to more complex arguments as well. However, creating minimal pairs becomes more complicated the more complex the arguments are.

[2] Some annotation schemes, e.g., Inference Anchoring Theory [6] distinguish between locutions and propositional content, when defining the building blocks of an argument. There, the propositional content is reconstructed from the original locution enriching it semantically, e.g., by resolving anaphora.

have mainly been used to assess *objective* argument quality. [41] acknowledge the *subjectivity* of rating argument quality, but do not explore this further. Our approach is based on the efforts to evaluate how convincing arguments are by [16,36,37] and similar approaches, and we treat argument preference as a single value derived from a function over linguistic features values of that argument.

We collect pairwise comparisons of arguments to train models that learn this function (e.g., [16]). More concretely, our approach is based on [36] (also [16,43]). This means we train argument preference models based on Gaussian Process Preference Learning (GPPL). We chose this model since it is particularly well suited to working with sparse data. Furthermore, [37] opens up new possibilities for future research by simultaneously including annotations from multiple users. As we are primarily interested in the impact of linguistic features on model performance, we focus on using linguistic features for assessing argument preferences, e.g., [5,29,44] to train these models.

Based on the collected argument preferences and the models trained on them, we can develop *user profiles* that explain the linguistic preferences of users. For this, two strategies are pursued in this paper: i) analyzing the feature importance scores that a model assigns during training, and ii) analyzing the most and least preferred arguments of a user using register analysis methods [4].

2.3 Visual Analytics for Linguistics

We integrate diverse methodologies from the domain of Visual Analytics [24] to support argument and model exploration as well as user engagement in the procedural stages of the CUEPipe. We draw upon expertise from prior studies in the field of natural language exploration [5,11]. Specifically, we derive methodologies from the field of visual data collection [12,25,33] to support the process of corpus annotation. We further integrate a new visual interactive labeling component derived from [2,3,34] for annotating argument preferences. Finally, we introduce a dashboard designed for the examination of preference models by introducing a new radial evaluation technique based on former approaches to user-centric visualization [8,18,34,35], thus adding to the growing body of work on LingVis: Visual Analytics for Linguistics [1,7].

3 The CUEPAQ Argument Exploration Pipeline

Our CUEPipe is a web-based application providing graphical user interfaces for various tasks related to the linguistic modeling of argument preferences. In this section, after introducing the overall workflow, we present the individual components, describing their basic functionalities and intended applications.

3.1 The CUEPipe Workflow

Figure 1 shows the overall workflow of the system. CUEPipe provides various interfaces (I) for working on collecting argument data. The (V)isualizations

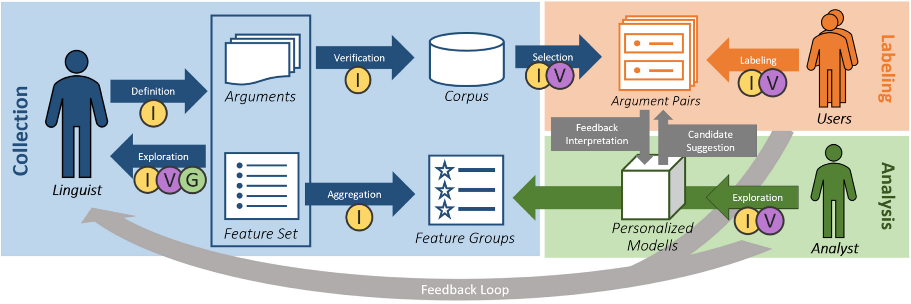


Fig. 1. The CUEPipe workflow

described in Sect. 3.2 provide an intuitive overview of the data set, allowing for its *exploration*. As described in Sects. 3.3 and 3.4, the labeling process and exploration of argument preferences are also supported by separate interfaces and visualizations.

Furthermore, the workflow in Fig. 1 highlights the different roles of entities interacting with the CUEPipe. It provides access to an extendable argument corpus. However, it is best used to study specific linguistic cues in a targeted data set. Thus, the first important role is that of the *linguist*. The linguist formulates a hypothesis and defines an expected outcome of the study. Then they generate a data set accordingly. Correspondingly, they may choose to specify a feature set that focuses on the attributes of interest.

The next step is conducting the study. The second role, *users*, consists of the target group. Here the subjective nature of argument preferences comes into play. The user group of a study can be categorized across different dimensions, e.g., demographic features, such as age, gender, or income. This depends on the goal of the study and the corresponding hypothesis. The task of the user group is to compare arguments pairwisely to create a model that captures their argument preferences reasonably well, as described in Sect. 3.3.

Finally, the role of the analyst is to interpret the resulting preference models and the insights they provide on the user group, e.g., finding clusters. The analyst has a dual role, as it should inform both the linguist and the users. Concerning the users, the goal is to teach them about their argument preferences by analyzing the features that play a role in their preference models and comparison with other models. With respect to the linguist, the analysis needs to communicate the actual outcome of the study, involving information about model performance and other factors that might affect the reliability of the study. This forms a feedback loop. Depending on the study's outcome, the linguist may want to revise their hypothesis or tweak other variables, such as the used feature space or the argument set. If the result confirms the hypothesis, the linguist still needs to evaluate the created models carefully to ascertain that the results are reliable.

The best use for the CUEPipe may be for prototyping studies to make sure that a more detailed investigation is warranted. However, it also allows linguists to expand on a study incrementally. In principle, the different elements are modular, allowing for individual use, too.

Table 1. Argument distribution in the CAP

Dataset	Arguments	Variation Ratio	Unique Standpoints
Corpus	315	0.69	78
Staging	1375	0.54	154
New	103	0.45	45

In the next few sections, we will present the individual steps in Fig. 1, *collection*, *labeling*, and *Analysis*, in more detail focusing on their implementation.

3.2 Generating a Data Set for Exploring Argument Preferences

The CUEPipe provides a graphical user interface for adding arguments to the Comparable Argument Corpus (CAP) we have developed. The main innovation of the CAP is that it allows adding minimal variations of arguments that contain contrasting lexical items. Thus, the interface is designed to provide a view for adding arguments, a view for varying arguments, and a general argument view that groups arguments and their variations to provide a high-level overview.

Data Collection: The corpus is divided into three levels, *new arguments*, *staging arguments*, and *corpus*. This distinction is mainly for quality control reasons. Arguments, as well as their variations, must adhere to the general structure described in Sect. 2.1: $(premise, conclusion, relation)$, arguments must be linguistically adequate (i.e., no non-sense strings, etc.), and the relation between *premise* and *conclusion* must be conceivable (thus, all arguments are assumed to surpass a certain argument quality threshold). After submission to *new arguments*, two additional data collectors have to confirm these requirements by promoting arguments to *staging* and *corpus*, respectively. Consequently, three distinct experts confirm each argument to be suitable for the corpus.[3] Table 1 describes the current size of the corpus. Variation ratio refers to the average number of variations per argument. *Unique standpoints* refers to the number of unique conclusions, indicating topic variation in the corpus. Since the goal is to focus on linguistic feature effects on argument preferences, we aim to provide a varied data set that allows the creation of test sets for various topics.

Each argument is annotated with metadata, including relations between arguments (i.e., whether an argument is a variation of another or an original argument), the *author* label for variations, and the *source* label for original

[3] Experts are linguistic researchers of argumentation and linguistics students with training in the annotation of arguments.

arguments.[4] For the sake of keeping the structure simple, there is no nesting of variations in the corpus, so variations generally have 0 other variations (although they may be incidental variations of other variations of the original).

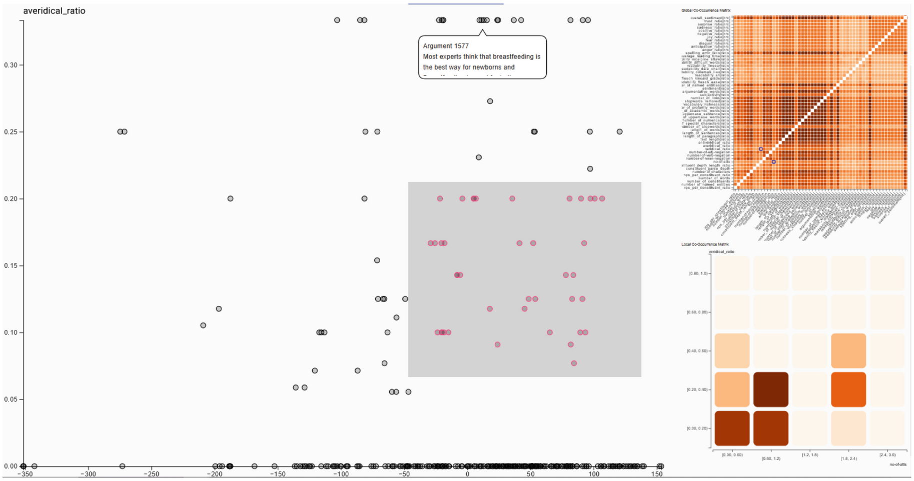
Linguistic Feature Annotations: Each argument in the main corpus is annotated with linguistic features to allow for the exploration of personalized argument preferences. For this, we use several automated feature annotation pipelines. Some of these were borrowed by other work, e.g., [11,39] and [29], while some features have been implemented actively for the CUEPipe. Particularly relevant for the CUEPipe are features introduced by lexical items, including the concrete use of certain items and additional properties. Examples of this are embedding verbs, noun and verb modifiers, and different types of negation (verb vs. noun). As an example of additional annotations related to the concrete lexical items, we use the semantic parser by [21] to distinguish different kinds of intensionality (veridical, averidical, and anti-veridical). Overall, the application supports 66 linguistic features, ranging from stylistic to semantic. These features are organized into feature groups that give an intuitive understanding of their expected role in analyzing personal argument preferences.

Corpus Exploration: In addition to the corpus management functionality, we provide a visual exploration dashboard to interact with the data in the corpus. This component primarily serves to inspect feature distributions and interactions in the corpus. It consists of three parts: the *argument similarity map* and the global and local *co-occurrence matrices.*

The *argument similarity map*, as the name suggests, maps arguments onto a two-dimensional space as circles. It distributes them according to their similarity based on their annotated features. For this, we use an off-the-shelf dimensionality reduction (principal component analysis, PCA; [17]) to reduce the linguistic feature vectors to two dimensions. The map can be customized for selected feature combinations. Thus, distributions of different feature categories based on the analyst's interest can be evaluated in this way. Moreover, different feature sets, or individual features, can be mapped onto the x- and y-axes of the map. As shown in Fig. 2, the selected features for each axis are reduced to one dimension each. This allows linguists to compare the distribution of features or feature groups in relation to the overall complexity of the corpus. Figure 2 illustrates this by presenting the distribution of the feature *averidical-ratio* relative to the full feature set. As the picture suggests, many arguments do not indicate averidicality in the selected argument set. However, of those marked with averidicality, we can see that they are somewhat distributed across the data set.[5] Linguists

[4] The arguments in the CAP have been collected from various sources, including existing argument corpora, e.g., the IBM 30k corpus [15] and the argumentative microtext corpus [28]. However, many arguments have also been collected manually with support from the OVA tool [19].

[5] Averidicality refers to elements that are linguistically marked as not factual, e.g., introduced by verbs like *believe*, *think*, *assume*, compared to verbs like *know*, *discover*, *forget*, which presuppose that the content that they mark is *veridical.* See Sect. 4.



Argument 1577
Most experts think that breastfeeding is
the best way for newborns and

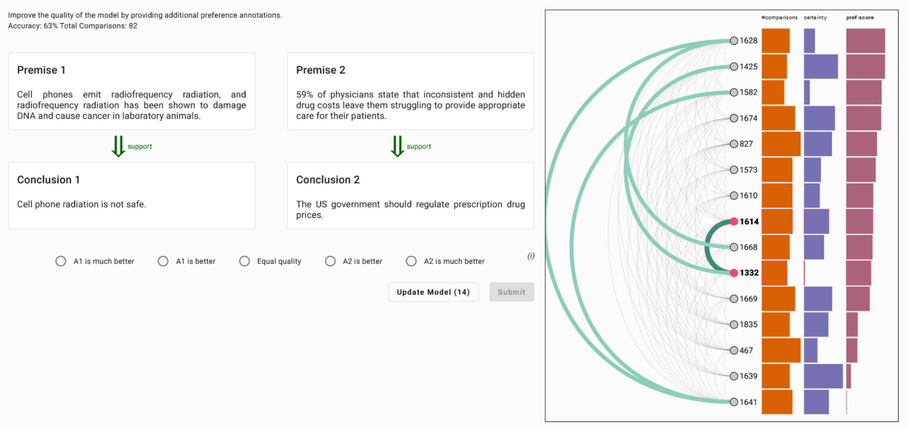
Fig. 2. Feature exploration

can select arguments of interest, such as argument clusters, outliers, or arguments of a certain value, for closer inspection to refine the information provided by the argument similarity map. As shown in Fig. 2 on the right, researchers then see global and local *feature co-occurrence matrices*. As the name suggests, these visualizations present feature collocations. The global matrix displays pairwise interactions within the selected subspace in the upper right corner. Darker shades indicate a high number of feature co-occurrences, while brighter shades indicate fewer feature co-occurrences. When a cell is selected, the local matrix (bottom right corner) shows how two features interact in close detail using the same overall method. Thus, the local co-occurrence matrix in Fig. 2 suggests that, in this selection, many arguments contain one propositional attitude verb expressing a level of veridicality.

Overall, the argument exploration dashboard can be used to find balanced data sets for specific features and to explore and reduce imbalances in test sets. Furthermore, it provides an overview of the coverage of the corpus.

3.3 Learning Preferences via Visual Interactive Labeling

Our goal is to learn preferences from pairwise comparisons, as illustrated in Fig. 3. There, two different arguments are presented. In accordance with our definition of an argument, choosing the preferred argument involves choosing the argument for which the premise better affects the conclusion (i.e., increases or lowers the acceptability of the conclusion). This task can be varied across various dimensions, e.g., by only presenting premises affecting the same conclusion, or only arguments with support relations, etc. Thus, the system allows for some flexibility concerning the definition of comparison tasks.



Improve the quality of the model by providing additional preference annotations.
Accuracy: 63% Total Comparisons: 82

Premise 1

Cell phones emit radiofrequency radiation, and radiofrequency radiation has been shown to damage DNA and cause cancer in laboratory animals.

Conclusion 1

Cell phone radiation is not safe.

Premise 2

59% of physicians state that inconsistent and hidden drug costs leave them struggling to provide appropriate care for their patients.

Conclusion 2

The US government should regulate prescription drug prices.

○ A1 is much better ○ A1 is better ○ Equal quality ○ A2 is better ○ A2 is much better

Fig. 3. Visual interactive labeling

The annotation of argument preferences is an extremely expensive task due to the fact that the number of comparisons n in a set of arguments with size x exhibits quadratic growth $(\mathrm{n} = (\,(x * (x - 1))/2)$. Thus, a full annotation of 30 arguments already requires 435 comparisons. Because we want to test personalized argument preferences, we cannot use multiple annotators for the same model to reduce the annotation cost per annotator. Consequently, we have developed a system that is aimed at supporting this costly annotation process and possibly reducing the number of annotations needed to make valid predictions about a user's argument preferences.

Learning Argument Preferences: For learning preferences, we represent arguments as linguistic feature vectors based on the annotations explained in Sect. 3.2. As an underlying model, we use a model for pairwise preference learning based on Gaussian Process Preference Learning [36,37], a type of Bayesian inference model. These models define a real-valued function f that takes linguistic feature values as input and can be used to predict rankings, pairwise labels, and ratings for individual arguments [36]. Concretely, ratings are represented as numeric values provided by f, where higher values correspond to a stronger preference for the given argument based on its features. Pairwise labels are predicted via the *preference likelihood* $p(i \succ j | f(x_i), f(x_j))$, where $i \succ j$ is a pairwise label comparing two arguments (i.e., argument i is better than argument j).

The application does not hinge on this choice of model. However, preliminary tests have shown the model's suitability for testing the overall pipeline. We primarily relied on its good performance on sparse data, allowing it to learn from relatively few comparisons, making it more feasible to learn the preferences of individual users.

Visual Interactive Labeling: The visual interactive labeling process is divided into two parts. First, a small random subset of comparisons is sampled from the data set that is to be annotated. A user annotates this subset to provide some initial comparisons for model training. Once the subset is fully annotated, the second stage begins.

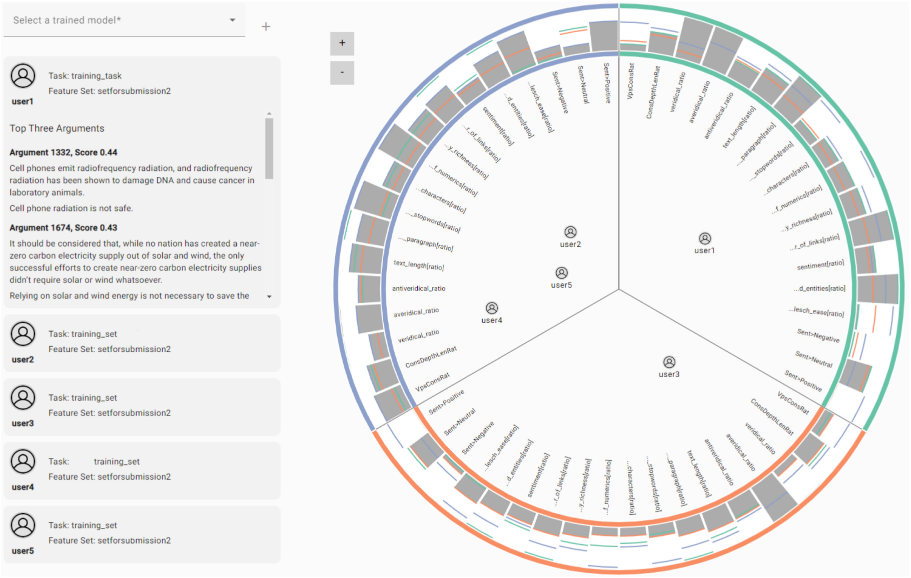
In the second stage, the user is supported by information from their preference model. Figure 3 illustrates our interface for visualizing model information. On the left-hand side, the two arguments are presented side-by-side. They are compared on a 5-point scale corresponding to the position of the arguments (i.e., A1 is the left argument, and A2 is the right argument). The visualization on the right side guides users through the annotation process. It can be divided into two parts divided by the arguments (represented by their IDs) as the spine of the visualization. On the left side, an arc diagram provides information about the overall annotation progress by visualizing the already annotated argument pairs in gray. Additionally, the arc diagram visualizes predictions by the user's model: the five green arcs suggest candidates for the next comparison based on the model's variance predicted for these comparisons. These suggestions are calculated globally across all arguments by default. The current arguments displayed for comparison are highlighted in pink as the comparison most favored by the model. The user can change the next pair of arguments by selecting other green arcs or clicking on single arguments. This action can become relevant when including argument-specific information in the decision process. As displayed on the right side, each argument is represented as a tuple of bar charts describing its number of annotated comparisons in orange, its assessment of the certainty of this score in blue, and its predicted absolute preference score in red. Sometimes, relying on the variance alone leads to a situation where only a certain subset of arguments is frequently annotated while other arguments are not annotated at all. Users may wish to strive for a more balanced annotation process. The visualization gives them the flexibility to do this. The visualization also allows users to investigate their annotation process by showing them the predicted ranking of the arguments based on their model. Thus, in addition to the concrete display of the accuracy value of the model, users can also confirm that the model learns the appropriate expected rankings for individual arguments.

Ultimately, this visualization serves to investigate strategies to quickly increase model accuracy, particularly during the annotation of large data sets. Once a large number of arguments is involved, it becomes unfeasible to annotate them all. Thus, doing the right annotations to increase model predictability is essential. As of now, we rely only on data from the trained models; However, the issue has been gaining more attention recently (e.g., [13]). Thus, future work aims at improving the model's capability to select meaningful comparisons.

3.4 Exploring Personal Preferences

The CUEPipe provides functionality for exploring preference models based on the previous steps of the pipeline. Concretely, we provide functionality for model performance analysis and model comparison across different users.

Model Performance Analysis: The application allows users to apply models to arbitrary data sets, allowing users to test them on unseen data. For this, k-fold cross-validation is provided as well (for k = 5). This allows users to train the model on larger data sets involving both seen and unseen data, providing a more in-depth understanding of the performance of a user's model. Additionally, we have added functionality for re-calculating the model training history. This allows us to investigate the model's performance in relation to the annotation progress of a given data set.



Select a trained model*

user1
Task: training_task
Feature Set: setforsubmission2

Top Three Arguments

Argument 1332, Score 0.44

Cell phones emit radiofrequency radiation, and radiofrequency radiation has been shown to damage DNA and cause cancer in laboratory animals.

Cell phone radiation is not safe.

Argument 1674, Score 0.43

It should be considered that, while no nation has created a near-zero carbon electricity supply out of solar and wind, the only successful efforts to create near-zero carbon electricity supplies didn't require solar or wind whatsoever.

Relying on solar and wind energy is not necessary to save the

user2
Task: training_set
Feature Set: setforsubmission2

user3
Task: training_set
Feature Set: setforsubmission2

user4
Task: training_set
Feature Set: setforsubmission2

user5
Task: training_set
Feature Set: setforsubmission2

Fig. 4. Model exploration

Comparative Model Exploration: The main visualization is presented in Fig. 4. It allows for the exploration and comparison of user models according to their predicted preferences. Again, we make use of a principal component analysis to project high-dimensional feature importance vectors provided by user models on a two-dimensional, radial space. Hence, models that are displayed close together share similar feature importance vectors. We use this metric as an indicator of the impact of linguistic features on the prediction of argument preferences. Thus, different feature importance values indicate different argument preferences.

To illustrate these differences, we separate the space into multiple slices by displaying a visualization similar to a pie chart. The user may determine the number of slices. The individual pieces of the pie describe potential model clusters, i.e., models with similar feature importance patterns. The feature importance vectors of these models are aggregated and visualized in the outer ring of the visualization. This provides users with information on differences between the various clusters. Color is used to affiliate the user model to the respective arc and to display important feature differences in the outer ring, thus supporting the differentiation between the different model clusters.

The model comparison visualization allows an analyst to cluster users and find commonalities between their models. To further explore the models, it is possible to extract top and bottom arguments from the annotated data sets (and beyond, if model performance allows it) and feed into the previously presented argument exploration view (Sect. 3.2). There, feature distributions in the different sets (all, top, and bottom arguments) may be inspected.

4 Study: Propositional Attitudes

We conducted a proof-of-concept case study to evaluate the functionality of the system. The study consists of a linguist creating a data set for exploring the impact of propositional attitude verbs on argument preferences. Subsequently, users were asked to compare arguments to learn their preferences. Finally, the results were analyzed using the presented model exploration functionality.

Creating Data Sets: For this proof-of-concept study, we created a data set consisting of arguments containing propositional attitude verbs, a kind of embedding verb encoding the commitment of a source to the embedded content.

More concretely, the properties we are interested in relate to the intensional nature of (some) embedding verbs [9,26]. One important notion is factuality or factivity [22], which also receives regular attention in the computational literature (e.g., [32,40]).

For the sake of this paper, we understand factivity as a continuous value that describes the degree of commitment attributed to the content embedded under factivity markers, e.g., *discover* in (3-a) or *believe* in (3-b). However, as (3-b) illustrates, the source that the commitment is attributed to is also relevant. Thus, the fact that 3 out of 50 lawyers believe that piracy is theft is not a strong premise for the standpoint *piracy is theft* (cf. (3-b)); however, were it 47 out 50 lawyers, then despite the less strong commitment indicated by *believe* (compared to *discover*), the premise might still make a good support.

(3) a. After evaluating over 105 million data points from 30,000 U.S.-based Prodoscore users, *we* **discovered** that there was a five percent increase in productivity during the pandemic work from home period. $\rightarrow_{support}$ Companies should move towards a hybrid and remote working environment.

b. *Of the 50 lawyers who were interviewed, only three* **believed** that downloading or streaming digital content from pirate sources should be illegal and unacceptable.
$\rightarrow_{support}$ Piracy is not theft.

The test and training data sets were created by a linguist based on 88 arguments containing propositional attitude verbs in the corpus. The data set was skewed towards embedding verbs *claim*, *think*, *agree*, and *show*. This is illustrated in Fig. 5 (right side). The embedding verbs used in the 15 test arguments are shown on the left of Fig. 5.

Preference Learning Experiments: We tested five participants (i.e., *users*) for this study. All of them had an academic background (three student assistants and two postdocs). Each participant did two annotation sessions. In the first session, they annotated preferences based on ten arguments, resulting in 45 random comparisons. These later serve as test sets for the models trained on their training sets comprising 105 comparisons. Due to the sparseness of the data, we tested the model both on seen and unseen data. The results in Table 2 show that while the model learned argument preferences for some of the users relatively well within the seen data, applying the models to unseen data shows that they have not learned enough to make general predictions about the argument preferences of the users (tested by combining the training and test sets and applying k-fold validation with k=5, as provided by the application).

Table 2. Model performance (accuracy) across users

	seen data	unseen data	standard deviation
user1	0.67	0.37	0.15
user2	0.82	0.62	0.09
user3	0.56	0.41	0.04
user4	0.79	0.59	0.03
user5	0.79	0.54	0.11

Model Exploration: We fed the four models based on the annotation study into the model exploration dashboard. The dashboard shows that the three users with relatively high accuracy on the seen data form a cluster with respect to the feature importance values of their models. *User1* and *User3* formed their own clusters (see Fig. 4). We can see that the models with the best accuracy metrics generally have higher feature importance scores across features. This suggests that their preference patterns are more consistent with underlying linguistic features. However, comparing the cluster of three in isolation reveals considerable differences in the importance of argument features, suggesting that the models are still quite distinct. Concretely, for *User5*, positive and negative sentiment were important features for the model, but the semantic features of veridicality and averidicality (i.e., those pertaining to factivity) did not seem to play a role. Conversely, *User2* put focus on neutral sentiment, and the semantic features concerned with factivity were among the most important ones of their model.

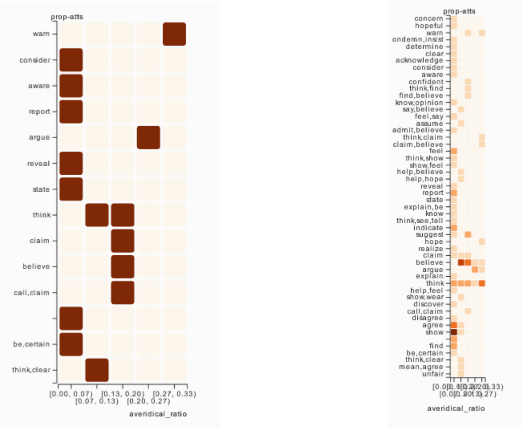


Fig. 5. Embedding verbs in the training set (left) and corpus (right)

Finally, *User5* model was mostly affected by features pertaining to linguistic complexity, while sentiment and factivity features played only a minor role.

Overall, the study's compactness requires us to take these assumptions with a grain of salt. Nonetheless, this proof of concept shows that the pipeline can be used to inspect personal argument preferences across multiple users within a few arguments. Informally collected feedback from the five users was very positive on average, although smaller technical issues occurred during the study. However, we will leave a more detailed analysis of the system's usability for future work.

5 Limitations

In this section, we discuss two limitations that pertain to the CUEPipe itself and to limitations of our proof-of-concept study.

5.1 The CUEPipe

Currently, CUEPipe is best suited for smaller pilot studies tailored toward the initial investigations of hypotheses. This limit is imposed on a technical level as well as on the level of implementation of the visualizations. On a technical level, the limit pertains mainly to the visual interactive labeling step. In the current implementation, model updates needed for making meaningful annotation suggestions require complete retraining of the model. This works well on smaller amounts of data but can interrupt the annotation process as the models become larger, both in terms of feature annotations and the number of comparisons. To some extent, this can be solved implementationally by optimizing training procedures (e.g., by running them asynchronously in the background and adapting the interface between the annotation interface and the trained preference models accordingly). Another possibility that could be explored is to create crowd models [37] and, for example, merge models of users with similar annotation behaviors. However, this would obviously lead to larger but (potentially) fewer models. Thus, ultimately, large-scale studies based on pairwise comparisons may require a more powerful infrastructure than is currently available.[6]

Another issue of the CUEPipe is that the visualizations do not always scale optimally with increasing data complexity. In particular, representing complex feature annotations can clutter visualizations. Thus, organizing and representing linguistic features intuitively is an ongoing concern. Our goal is to improve on the current state which only allows the selection and deselection of features. A more ambitious approach would be to incorporate guidance. For example, in the radial exploration visualization, such a system could attempt to automatically detect features relevant to distinguishing target groups and highlight those.

5.2 The Proof-of-concept Study

The study focused on the system's overall usability, concentrating on the workflow described in Sect. 3.1. As mentioned there, the study should be seen as a prototype study. The main drawbacks are as follows:

The study participants have not been selected with certain demographic properties in mind. Although the system can and should be used to find differences in seemingly homogenous groups (in this case, all participants were academics), a study that is geared towards predicting predefined clusters in a target group may illustrate the validity of the system more clearly.

Overall, the study is small-scale. Thus, as mentioned in Sect. 4, the results should be taken with a grain of salt. This is further compounded by the fact that we rely on automated feature annotations. While this is fine for some features, e.g., those pertaining to language complexity, in particular, the meaning-oriented features, such as veridicality, need to be carefully evaluated to avoid propagating wrong information in the analysis stage of the system. For example, the system broadly captures the right generalizations regarding the relation between attitude verbs and veridicality, but there exist some outliers that can falsify results. Concerning the first problem, future studies are planned with a focus on exploring the individual properties of target groups. Regarding the second problem, including evaluation metrics for features may make the system more transparent.

6 Conclusion

We have presented an application combining three major components for researching personalized argument preferences: data collection, preference labeling, and preference exploration. We also contribute a small (but dynamic) corpus of linguistically annotated arguments and various techniques for visual analysis of linguistic data. The CUEPipe application has been demonstrated by means of a proof-of-concept study, indicating that the overall workflow is successful.

[6] As [36,37] show, large-scale studies are not generally impossible for GPPL. The concerns here target the visual support component of the annotation task.

The pipeline offers up multiple avenues for future work, e.g., facilitating comparative annotation, the visual representation of linguistically annotated data, and the visual exploration of linguistic models. Overall, the CUEPipe provides exciting prospects for exploring personalized argument preferences. Its coverage of various major tasks in linguistic research makes it interesting for everyone working on argument preferences. Furthermore, its ease of use reduces the barrier to conducting various tasks for users new to the topic.

References

1. Beck, C., Butt, M.: Visual analytics for historical linguistics: opportunities and challenges. J. Data Min. Dig. Hum. Special issue Visualisat. Historical Linguist. 1–23 (2020)
2. Bernard, J., Zeppelzauer, M., Sedlmair, M., Aigner, W.: A Unified Process for Visual-Interactive Labeling. The Eurographics Association (2017). https://doi.org/10.2312/eurova.20171123, https://diglib.eg.org:443/xmlui/handle/10.2312/eurova20171123. Accessed 12 Jun 2017. T05:16:33Z
3. Bernard, J., Zeppelzauer, M., Sedlmair, M., Aigner, W.: VIAL: a unified process for visual interactive labeling. Vis. Comput. **34**(9), 1189–1207 (2018). https://doi.org/10.1007/s00371-018-1500-3
4. Biber, D., Conrad, S.: Register, Genre, and Style. Cambridge University Press, Cambridge (2009)
5. Bögel, T., et al.: Towards Visualizing Linguistic Patterns of Deliberation: a Case Study of the S21 Arbitration (2014). talk presented at DH2014 in Lausanne
6. Budzynska, K., Janier, M., Reed, C., Saint-Dizier, P., Stede, M., Yakorska, O.: A model for processing illocutionary structures and argumentation in debates. In: Proceedings of the 9th edition of the Language Resources and Evaluation Conference (LREC) (2014)
7. Butt, M., Hautli-Janisz, A., Lyding, V. (eds.): LingVis: Visual Analytics for Linguistics. CSLI Publications, Stanford (2020)
8. Cashman, D., et al.: A user-based visual analytics workflow for exploratory model analysis. Comput. Graphics Forum **38**(3), 185–199 (2019). https://doi.org/10.1111/cgf.13681, https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.13681, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13681
9. Condoravdi, C., Crouch, D., De Paiva, V., Stolle, R., Bobrow, D.: Entailment, intensionality and text understanding. In: Proceedings of the HLT-NAACL 2003 Workshop on Text Meaning, pp. 38–45 (2003)
10. DeKeyser, R.M.: The robustness of critical period effects in second language acquisition. Stud. Second. Lang. Acquis. **22**(4), 499–533 (2000)
11. El-Assady, M., et al.: lingvis.io - A Linguistic Visual Analytics Framework (2019)
12. Van den Elzen, S., Van Wijk, J.J.: BaobabView: interactive construction and analysis of decision trees. In: 2011 IEEE Conference on Visual Analytics Science and Technology (VAST), pp. 151–160 (2011). https://doi.org/10.1109/VAST.2011.6102453, https://ieeexplore.ieee.org/document/6102453
13. Gienapp, L., Stein, B., Hagen, M., Potthast, M.: Efficient pairwise annotation of argument quality. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 5772–5781 (2020)
14. Gold, V., Hautli-Janisz, A., Holzinger, K.: VisArgue - Analyse von Politischen Verhandlungen. Zeitschrift für Konfliktmanagement **3**(16), 98–99 (2016)

15. Gretz, S., et al.: A large-scale dataset for argument quality ranking: construction and analysis. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 7805–7813 (2020)
16. Habernal, I., Gurevych, I.: Which argument is more convincing? Analyzing and predicting convincingness of web arguments using bidirectional LSTM. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1589–1599 (2016)
17. Hasan, B.M.S., Abdulazeez, A.M.: A review of principal component analysis algorithm for dimensionality reduction. J. Soft Comput. Data Min. **2**(1), 20–30 (2021)
18. Hindalong, E., Johnson, J., Carenini, G., Munzner, T.: Abstractions for visualizing preferences in group decisions. Proc. ACM Hum.-Comput. Interact. **6**(CSCW1), 49:1–49:44 (2022). https://doi.org/10.1145/3512896, https://dl.acm.org/doi/10.1145/3512896
19. Janier, M., Lawrence, J., Reed, C.: OVA+: an argument analysis interface. In: Computational Models of Argument (COMMA) (2014)
20. Johnson, J.S., Newport, E.L.: Critical period effects in second language learning: the influence of maturational state on the acquisition of English as a second language. Cogn. Psychol. **21**(1), 60–99 (1989)
21. Kalouli, A.L., Crouch, R., de Paiva, V.: GKR: bridging the gap between symbolic/structural and distributional meaning representations. In: Xue, N., et al. (eds.) Proceedings of the First International Workshop on Designing Meaning Representations, Florence, Italy, pp. 44–55. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/W19-3305, https://aclanthology.org/W19-3305
22. Karttunen, L.: Some observations on factivity. Res. Lang. Soc. Interact. **4**(1), 55–69 (1971)
23. Katzav, J., Reed, C.: On argumentation schemes and the natural classification of arguments. Argumentation **18**(2), 239–259 (2004)
24. Keim, D., Andrienko, G., Fekete, J.-D., Görg, C., Kohlhammer, J., Melançon, G.: Visual analytics: definition, process, and challenges. In: Kerren, A., Stasko, J.T., Fekete, J.-D., North, C. (eds.) Information Visualization. LNCS, vol. 4950, pp. 154–175. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-70956-5_7
25. Lu, Y., Wang, H., Landis, S., Maciejewski, R.: A visual analytics framework for identifying topic drivers in media events. IEEE Trans. Visual. Comput. Graphics **24**(9), 2501–2515 (2018). https://doi.org/10.1109/TVCG.2017.2752166, https://ieeexplore.ieee.org/document/8037991, conference Name: IEEE Transactions on Visualization and Computer Graphics
26. Nairn, R., Condoravdi, C., Karttunen, L.: Computing relative polarity for textual inference. In: Proceedings of the fifth International Workshop on Inference in Computational Semantics (ICOS-5) (2006)
27. Park, K., Park, M.K., Song, S.: Deep learning can contrast the minimal pairs of syntactic data. Linguistic Res. **38**(2), 395–424 (2021)
28. Peldszus, A., Stede, M.: An annotated corpus of argumentative microtexts. In: Proceedings of the First Conference on Argumentation, Lisbon, Portugal (2016)
29. Plenz, M., Buchmüller, R., Bondarenko, A.: Argument quality prediction for ranking documents. In: Working Notes Papers of the CLEF 2023 Evaluation Labs, CEUR Workshop Proceedings (2023)
30. Rahwan, I., Reed, C.: The argument interchange format. In: Rahwan, I., Reed, C. (eds.) Argumentation in Artificial Intelligence. Spring, Cham (2009). https://doi.org/10.1007/978-0-387-98197-0_19

31. Reed, C., Walton, D.: Argumentation schemes in dialogue. In: Hansen, H.V., et al. (eds.) Dissens and the Search for Common Ground, pp. 1–11, Windsor, ON. OSSA (2007)
32. Saurí, R., Pustejovsky, J.: Are you sure that this happened? Assessing the factuality degree of events in text. Comput. Linguist. **38**(2), 261–299 (2012)
33. Schmid, J., Cibulski, L., Hazwani, I.A., Bernard, J.: RankASco: a visual analytics approach to leverage attribute-based user preferences for item rankings. In: Bernard, J., Angelini, M. (eds.) EuroVis Workshop on Visual Analytics (EuroVA). The Eurographics Association (2022). https://doi.org/10.2312/eurova.20221072
34. Sevastjanova, R., El-Assady, M., Bradley, A., Collins, C., Butt, M., Keim, D.: VisInReport: complementing visual discourse analytics through personalized insight reports. IEEE Trans. Visual. Comput. Graph. **28**(12), 4757–4769 (2022). https://doi.org/10.1109/TVCG.2021.3104026
35. Sevastjanova, R., Hauptmann, H., Deterding, S., El-Assady, M.: Personalized language model selection through gamified elicitation of contrastive concept preferences. IEEE Transa. Visual. Comput. Graphics 1–17 (2023). https://doi.org/10.1109/TVCG.2023.3296905, https://ieeexplore.ieee.org/document/10194961, conference Name: IEEE Transactions on Visualization and Computer Graphics
36. Simpson, E., Gurevych, I.: Finding convincing arguments using scalable Bayesian preference learning. Trans. Assoc. Comput. Linguist. **6**, 357–371 (2018)
37. Simpson, E., Gurevych, I.: Scalable Bayesian preference learning for crowds. Mach. Learn. **109**(4), 689–718 (2020)
38. Sperrle, F., Sevastjanova, R., Kehlbeck, R., El-Assady, M.: Viana: visual interactive annotation of argumentation. In: Proceedings of IEEE Conference on Visual Analytics Science and Technology (VAST), pp. 12 (2019). https://doi.org/10.1109/VAST47406.2019.8986917
39. Sperrle, F., Sevastjanova, R., Kehlbeck, R., El-Assady, M.: VIANA: visual interactive annotation of argumentation. CoRR abs/1907.12413 (2019). http://arxiv.org/abs/1907.12413
40. Stanovsky, G., Eckle-Kohler, J., Puzikov, Y., Dagan, I., Gurevych, I.: Integrating deep linguistic features in factuality prediction over unified datasets. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 352–357 (2017)
41. Toledo, A., et al.: Automatic argument quality assessment–new datasets and methods. arXiv preprint arXiv:1909.01007 (2019)
42. Van Eemeren, F.H., Grootendorst, R., Johnson, R.H., Plantin, C., Willard, C.A.: Fundamentals of Argumentation Theory: A Handbook of Historical Backgrounds and Contemporary Developments. Routledge (2013)
43. Wachsmuth, H., et al.: Argumentation quality assessment: theory vs. practice. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 250–255 (2017)
44. Wachsmuth, H., et al.: Computational argumentation quality assessment in natural language. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Valencia, Spain, pp. 176–187. Association for Computational Linguistics (2017). https://www.aclweb.org/anthology/E17-1017
45. Wachsmuth, H., Werner, T.: Intrinsic quality assessment of arguments. arXiv preprint arXiv:2010.12473 (2020)
46. Walton, D., Macagno, F.: A classification system for argumentation schemes. Argument Comput. **6**(3), 219–245 (2015)

47. Walton, D., Reed, C., Macagno, F.: Argumentation Schemes. Cambridge University Press, Cambridge (2008)
48. Warstadt, A., et al.: BLiMP: the benchmark of linguistic minimal Pairs for English. Trans. Assoc. Comput. Linguist. **8**, 377–392 (2020)

Argument Search and Retrieval

Extending the Comparative Argumentative Machine: Multilingualism and Stance Detection

Irina Nikishina[1], Alexander Bondarenko[2](✉), Sebastian Zaczek[1], Onno Lander Haag[2], Matthias Hagen[2], and Chris Biemann[1]

[1] Universität Hamburg, Hamburg, Germany
[2] Friedrich-Schiller-Universität Jena, Jena, Germany
alexander.bondarenko@uni-jena.de

Abstract. The comparative argumentative machine CAM can retrieve arguments that answer comparative questions—questions that ask which of several to-be-compared options should be favored in some scenario. In this paper, we describe how we equipped CAM with a better answer stance detection (i.e., a better detection of which option "wins" a comparison) and with system variants to support non-English requests. As for the improved answer stance detection, we develop RoBERTa-based approaches and experimentally show them to be more effective than previous feature-based and LLM-based stance detectors. As for the multilingualism, in a proof of concept, we compare two approaches to support Russian requests and answers: (1) translating the original English CAM data and (2) using an existing replica of CAM on native Russian data. Comparing the translation-based and the replica-based CAM variants in a user study shows that combining their answers seems to be the most promising. For individual questions, the retrieved arguments of the two variants are often different and of quite diverse relevance and quality. As a demonstrator, we deploy a first multilingual CAM version that combines translation-based and replica-based outputs for English and Russian and that can easily be extended to further languages.

Keywords: Answering Comparative Questions · Argumentation Machines · Answer Stance Detection · Cross-Language Argument Retrieval

1 Introduction

Decision making is part of everyday life, yet it can involve a complex and time-consuming process when pro / con arguments on the potential alternatives need to be gathered and weighed (e.g., 'Should I buy or rent a house?').

There are many ways to gather arguments or opinions on some comparison objects: asking other people but also using web search engines, LLM-based

I. Nikishina and A. Bondarenko—These authors contributed equally.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 317–334, 2024.
https://doi.org/10.1007/978-3-031-63536-6_19

systems, specialized product comparison websites, research prototypes, etc. One research prototype that was developed for open comparative questions (i.e., not just focusing on products) and that should be more effective than skimming through a search engine's classical "ten blue links" is the comparative argumentative machine CAM [33].[1] The web interface of CAM takes some comparison objects and aspects as input, retrieves (argumentative) sentences relevant to the comparison, detects the sentences' comparative stances (i.e., which comparison object is favored), and presents a tabular result. As the original CAM only supports English inputs and results, recently, a replica of CAM for Russian comparisons—the RuCAM system—has been developed [22].

An important component of CAM and RuCAM is stance detection to determine for each retrieved sentence which comparison object is favored and which object is the overall "winner" in the retrieved sentences (e.g., 'buying' vs. 'renting' in the house example). Accurately grouping the retrieved sentences in CAM's and in RuCAM's tabular result presentation with respect to the favored objects ensures that users are not misled and can come to "correct" conclusions. Still, with F1 scores of 0.85 and 0.82, respectively, CAM's and RuCAM's current rule- and XGBoost-based [8] or rule- and BERT-based [13] stance detection seem to leave some room for further effectiveness improvement. Our first research question thus is: **(RQ1)** Can advanced BERT models like RoBERTa improve the effectiveness of CAM's and RuCAM's answer stance detection?

To address RQ1, we fine-tune several RoBERTa-based models [9,20] on the 5,759 English sentences of the CompSent-19 training set [29] using the masking approach of Bondarenko et al. [4]. In our experiments, our new stance detectors achieve F1 scores of 0.91 for English and 0.87 for Russian and thus are more effective than CAM's and RuCAM's current detectors. Interestingly, a multilingual XLM-RoBERTa model turned out to be as effective as an English-only model so that the same stance detector can be used in CAM and in RuCAM.

Developed as a replica of the original CAM system, RuCAM uses the Russian and not the English part of the Common Crawl.[2] Still, another possibility to support some non-English language would have been to simply translate the original CAM's inputs and results. As there was no comparison of these two ideas yet and as we aim for a single multilingual CAM system, our second research question is: **(RQ2)** What are the strengths and weaknesses of machine translation-based and replica-based "localization" of CAM?

To address RQ2, we use Russian as the target language and compare machine-translated CAM responses and RuCAM responses in manual analyses and in a user study. The results of our manual analyses indicate that CAM and RuCAM retrieve rather different results of varying relevance and quality (translated CAM results tend to be more relevant while the RuCAM results tend to be of higher quality). Therefore, a combination of translation-based and replica-based results seems to be a promising direction for a multilingual CAM system.

[1] http://ltdemos.informatik.uni-hamburg.de/cam/.

[2] https://commoncrawl.org/.

Our code and data are publicly available.[3] As a demonstrator of a multilingual CAM system, we equip the existing CAM and RuCAM backends with a new interface for multilingual translation- and replica-based search in English and Russian[4] that can easily be extended to support further languages.

2 Related Work

This section provides an overview of the CAM system and the existing approaches for comparative stance detection.

2.1 CAM Overview

Comparative information needs in web search were first addressed by developing simplistic search interfaces where two to-be-compared products were entered separately in a left and a right search box [25,35]. The search results were presented to the searcher as side-by-side two standard "ten blue links" lists for each product. Later, the comparative argumentative machine (CAM) was developed to tackle open-domain comparisons (not just products) [33]. The CAM's web interface takes as input user-specified two comparison objects (e.g., 'buy a house' and 'rent a house') and optional comparison aspect(s) (e.g., 'risk'). Then, using Elasticsearch,[5] CAM retrieves comparative arguments (e.g., 'It is less risky to rent a house than to buy') relevant to the user input from the DepCC corpus [30] containing about 14 billion English sentences coming from the Common Crawl corpus (if no relevant arguments are found, CAM will respectively notify the user). For each retrieved argument, its comparative stance is detected so that the arguments can be grouped into two columns (i.e., whether one or the other object is preferred according to individual arguments) for the final result presentation. In the CAM's output, the arguments are ranked based on the Elasticsearch relevance score. Additionally, the final comparison score is shown to the user, which determines the overall "winner" of the comparison. The score combines the stance detector's confidence and the Elasticsearch score and is summed up over all the retrieved arguments for each comparison object [33]. The design of the CAM system is also shown in Fig. 1.

The comparative stance in the context of CAM is defined as ternary label: (1) First comparison object "wins" a comparison, i.e., it performs better or is more suitable for the comparison aspect compared to the second object (e.g., 'It is less risky to buy a house than to rent a house'), (2) second object "wins" a comparison (e.g., 'It is less risky to rent a house than to buy a house'), or (3) none "wins" or no comparison is present; such statements are excluded from the CAM's final result presentation [33].

A user study with CAM showed that the study participants were able to answer comparative questions faster and more accurately compared to a standard

[3] https://github.com/webis-de/RATIO-24.

[4] https://cam-multi.ltdemos.informatik.uni-hamburg.de.

[5] https://www.elastic.co.

web search [33]. Later, RuCAM [22], a Russian version of the CAM system (that supports the English language only), was developed that replicates the original CAM pipeline using a Russian part of the Common Crawl corpus [28].

2.2 Comparative Stance Detection

One of the important (Ru)CAM components is a comparative stance detector that allows to place the retrieved arguments on the correct "winning" side. More generally, stance detection is the task of identifying the author's viewpoint (attitude, opinion) towards a target, which can be a debate topic, an entity, or a claim [23]. Earlier works mostly focused on the stance detection in online debates [19] and proposed rule-based [2,24,41,42] and feature-based classifiers using, for instance, SVM [3,15,36,42] or Naïve Bayes [2,15,31]. Later, neural network architectures like CNN [16,43], LSTM [44,45], and transformer-based models like BERT [13] became state-of-the-art approaches [1,34,39].

The aforementioned works mostly focused on the stance detection towards a single target. Our task is to detect the stance given two comparison objects, i.e., two stance targets (e.g., 'buy a house' vs. 'rent a house'). For detecting a comparative stance, i.e., a "winning" comparison object in English sentences, different feature-based classifiers were tested [29], e.g., logistic regression [12], SVM [11], XGBoost [8], etc. Trained on 5,759 and tested on 1,440 English sentences (CompSent-19 dataset [29]), the most effective stance detector (XGBoost with InferSent embeddings [10]) achieved a micro-avg. F1 of 0.85 (3 labels: first / second object "wins" or no comparison).

Later, on the same dataset, a stance detector was tested that employed multi-hop graph attention over a dependency graph sentence representation [21]. Each sentence was represented by its dependency graph, which, for simplicity, was then converted from the original directed graph into an undirected graph. Embeddings for each sentence word (node in the dependency graph) were calculated using BERT [13]. Then, Graph Attention Networks [40] were used to embed the relation between the comparison objects. Finally, a feed-forward layer with a softmax function was added to project the embedding vectors into classes for prediction. The proposed approaches achieved a micro-avg. F1 of 0.87, outperforming the previous XGBoost-based stance detector.

Recently, large language models like LlaMa-2 [38], GPT-3.5 Turbo [26], and GPT-4 [27] using zero-shot and few-shot prompting were tested [18]. In addition to rigorous prompt engineering, the authors designed a retry message to tackle the cases when an LLM returned malformed answers, i.e., answers violating the predefined format suitable to extract the predicted stance. Interestingly, all tested LLMs did not improve over the aforementioned stance detectors.

In this paper, we fine-tune the RoBERTa [20] and XLM-RoBERTa models [9] following the idea of masking the comparison objects with special tokens [4]. The evaluation results show that our stance detectors are more effective than previous feature-, neural-, and LLM-based approaches, achieving a micro-avg. F1 of 0.91 for the English and 0.87 for the Russian languages.

3 Improving Comparative Stance Detection

The current CAM implementation allows to choose between a rule-based comparative stance detector that uses the handcrafted list of cue words and an XGBoost-based classifier [29,33]. The latter one is more effective and achieves a micro-avg. F1 of 0.85 (3 labels: first/second object "wins" or no comparison).

Table 1. Stance detection effectiveness of different approaches tested on English sentences from the CompSent-19 dataset (class distribution: 'no comparison' 73%, 'first object wins' 19%, 'second object wins' 8%) [29]. Reported are F1 scores per stance class and a micro-averaged F1.

Stance detector	Ref.	Stance label			Micro-avg.
		First	Second	None	
Rule-based	[29]	0.65	0.44	0.90	0.82
GPT-3.5 Turbo (few-shot)	[18]	0.68	0.48	0.90	0.84
LLaMa-2 70B (few-shot)	[18]	0.75	0.60	0.91	0.85
XGBoost + InferSent	[29]	0.75	0.43	0.92	0.85
GPT-4 (few-shot)	[18]	0.78	0.65	0.91	0.86
ED-GAT$_{BERT}$	[21]	0.78	0.56	0.93	0.87
RoBERTa-masked	our	0.86	**0.70**	0.94	**0.91**
XLM-RoBERTa-masked	our	**0.87**	0.69	**0.95**	**0.91**

To address our first research question, we fine-tune the English RoBERTa [20] and multilingual XLM-RoBERTa [9] models on the CompSent-19 train set (5,759 English sentences) [29]. Following the idea by Bondarenko et al. [4], we mask the comparison objects using the special masking tokens: `[FIRST OBJECT]` and `[SECOND OBJECT]`, before fine-tuning.[6]

We compare the effectiveness of our stance detectors with several approaches from previous work that were tested on the CompSent-19 test set (1,440 English sentences; we again mask the comparison objects). The results in Table 1 show that our stance detectors achieve a convincing micro-avg. F1 of 0.91 and are more effective than previous existing feature-based and LLM-based approaches.

Multilingual Stance Detection. To test our comparative stance detector for the Russian language, we use a dataset of 1,208 manually labeled Russian sentences [22]. Due to a relatively small number of labeled examples, we use the whole dataset to test our multilingual stance detector based on XLM-RoBERTa that was fine-tuned on English sentences. We also test on the original test set for a more fair comparison with the stance detectors for Russian from previous work [22]. Our stance detector achieves a micro-avg. F1 of 0.87 on the test

[6] Hyperparameters were selected using a 5-fold cross-validation on the train set. Models: roberta-large and xlm-roberta-large, batch size: 16, learning rate: 0.00003, training epochs: 5. Fine-tuning was performed using Colab's Tesla T4 GPU (RoBERTa) and NVIDIA GeForce RTX 4090 (XLM-RoBERTa).

Table 2. Stance detection effectiveness of different approaches tested on Russian sentences (class distribution: 'no comparison' 70%, 'first object wins' 21%, 'second object wins' 9%) [22]. Reported are F1 scores per stance class and a micro-averaged F1. For comparison with the original stance detectors for Russian [22], we additionally report the results on the original test set.

Stance detector	Ref.	Stance label			Micro-avg.
		First	Second	None	
Test set (119 sentences)					
Rule-based	[22]	0.34	0.33	0.82	0.69
RuBERT-based	[22]	0.57	0.38	0.91	0.82
XLM-RoBERTa-masked	our	**0.71**	**0.53**	**0.93**	**0.87**
Full dataset (1,208 sentences)					
Rule-based	[22]	0.41	0.27	0.74	0.62
XLM-RoBERTa-masked	our	**0.68**	**0.53**	**0.90**	**0.83**

set and 0.83 on the full dataset (cf. Table 2), which is more effective than the rule-based approach and fine-tuned RuBERT [22].

An interesting observation is that for English sentences, fine-tuning a multilingual RoBERTa is as effective for stance detection as fine-tuning an English-only model. We thus suggest using for practical application a single fine-tuned XLM-RoBERTa to detect the stance in both the English and Russian languages.

4 Adapting CAM to Russian

One of the limitations of CAM is its restriction on the English language. To address our second research question, we explore the use of machine translation to translate CAM output to the target Russian language. We then compare the translation-based approach with the existing replica-based system, RuCAM [22].

4.1 Translation-Based System

As the translation model, we use OPUS-MT [37]. While OPUS-MT could be replaced with any system, we motivate its use since it is open access and easy to implement using the Huggingface's transformers library.

The overall CAM architecture extended with the translation modules is presented in Fig. 1(a). First, we added a translation step to the CAM's input of the comparison objects and aspect(s) from Russian to English,[7] so that we could retrieve English sentences from the Elasticsearch index. Afterwards, we translate the retrieved arguments from English to Russian for the final result presentation.[8] This system does not require any additional data and indexing.

[7] https://huggingface.co/Helsinki-NLP/opus-mt-ru-en.

[8] https://huggingface.co/Helsinki-NLP/opus-mt-en-ru.

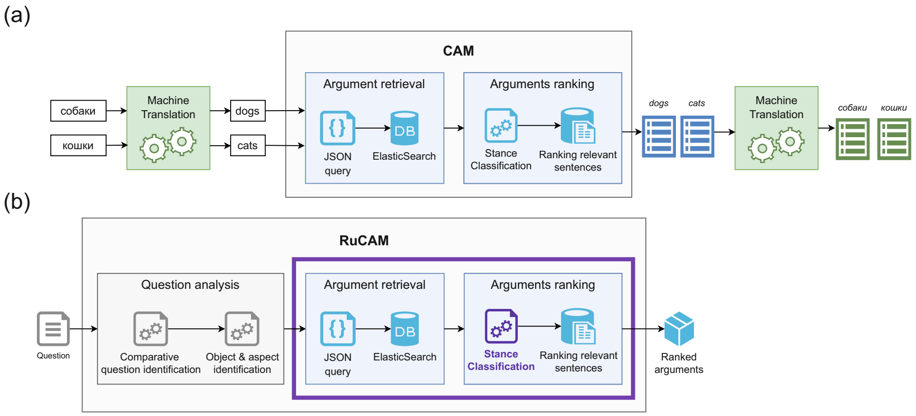


Fig. 1. (a) The architecture of the CAM system extended with translation modules; new modules are in green; (b) The architecture of the RuCAM system [22]. The modules that we used for the language adaptation comparison are in purple. The updated stance classification model is also highlighted in purple. (Color figure online)

4.2 RuCAM

To compare with the alternative pipeline for language adaptation, we consider RuCAM [22]: the Russian Comparative Argumentative Machine[9] that implements its own Elasticsearch index based on the Open Super-large Crawled Aggregated corpus (OSCAR) [28] containing 21 billion Russian sentences from the Common Crawl corpus. RuCAM also accepts natural language questions as input; however, we skip previous steps (comparative question identification and object and aspect detection) and query the system directly with two comparison objects and, optionally, aspect(s). We also replace the original stance classification model with our fine-tuned XLM-RoBERTa-masked model to improve the quality of the system output. Figure 1(b) presents the overall architecture of RuCAM, highlighting in purple the modules used for the comparison as well as the updated stance classification model.

4.3 System Comparison on Retrieval Effectiveness

To evaluate the two adaptation techniques, we set up a manual annotation following the methodology from the Touché 2020–2022 shared tasks on argument retrieval [5–7]. For the user study, we use the Touché 2022 dataset [6] that contains 50 comparative questions, each labeled with two comparison objects that we also translated into Russian (e.g., ‘Should I buy or rent a house?’) The English CAM found matches (i.e., relevant arguments) for 40 object pairs, and the RuCAM had matches for 46 pairs. Then, we provided five volunteer annotators with the annotation guidelines from the Touché tasks and asked to label the

[9] http://rucam.ltdemos.informatik.uni-hamburg.de.

Table 3. Relevance-wise (R) and quality-wise (Q) retrieval effectiveness of the replica-based and translation-based systems. The nDCG scores are calculated for the two ranked lists separately (the first or the second object "wins" a comparison split after stance detection) and for all the retrieved results before stance detection (overall).

System	First object				Second object				Overall			
	nDCG@5		nDCG@10		nDCG@5		nDCG@10		nDCG@5		nDCG@10	
	R	Q	R	Q	R	Q	R	Q	R	Q	R	Q
Replica-based	0.84	**0.93**	0.90	**0.96**	0.83	**0.93**	0.90	**0.96**	0.83	**0.93**	0.90	**0.96**
Transl.-based	**0.91**	0.81	**0.97**	0.91	**0.85**	0.79	**0.94**	0.90	**0.88**	0.78	**0.95**	0.91

retrieved arguments based on: (1) the relevance to the comparison object pairs as not relevant (label 0), relevant (label 1), and highly relevant (label 2), and (2) the argument quality (rhetorical well-writtenness) as low quality (label 0), sufficient quality (label 1), and high quality (label 2). In total, the labeled dataset comprises 1,238 arguments (at most 10 arguments for each object pair).[10]

To calibrate the annotators' interpretations of the guidelines, we conducted an initial Fleiss' κ test in which each annotator had to label the same 15 arguments for 3 object pairs (5 arguments for each pair). The observed Fleiss' κ values were 0.734 for argument relevance (substantial agreement) and 0.45 for argument quality (moderate agreement). Furthermore, after the initial κ test, we organized a follow-up discussion with all the annotators to clarify potential misinterpretations, e.g., the cases where one of the objects is ambiguous (e.g., 'Milk tastes better than cow's milk in the supermarket.'; it is not clear whether the first object refers to 'goat milk' or not) or the argument is too long and contains both the comparison as well as the unrelated text (e.g., 'Comp JAVA Program Description: I will not say for sure that NetBeans is better than Eclipse, I believe that each development environment has its own strengths and weaknesses.'). Afterwards, each annotator independently judged the results for disjoint subsets of the topics (i.e., each unique object pair was assigned to one annotator only).

Using the resulting manual labels for the argument relevance and quality, we calculate nDCG@5 and nDCG@10 scores [17] for each comparison object separately (since CAM and RuCAM split the arguments according to the "winning" object) and overall scores for all arguments retrieved for all object pairs (see Table 3). We assume that the scores are comparable, as both systems retrieve arguments from Common Crawl. Apparently, different corpora might have different numbers of relevant arguments because of social and cultural differences.

The results show that both relevance and quality nDCG scores are relatively high for both systems. However, the argument relevance in a translation-based system is consistently higher. This might be explained either by a more effective stance detector for English or a more effective Elasticsearch retrieval for English, possibly due to linguistic differences (Russian is a highly inflected language while

[10] The labeled dataset and annotation guidelines are available in the GitHub repository: https://github.com/webis-de/RATIO-24.

Table 4. Similarity scores between the arguments retrieved by the translation-based and replica-based systems. The respective embedding models used to calculate a cosine similarity are given in parentheses.

Metric	Score
ROUGE-1	0.257
ROUGE-2	0.046
Cos. sim. (sentence-transformers/LaBSE)	0.404
Cos. sim. (ai-forever/sbert_large_nlu_ru)	0.635
Cos. sim. (DeepPavlov/rubert-base-cased-sentence)	0.656

English is weakly inflected). On the other hand, the argument quality-wise nDCG scores for translated arguments are lower than those for arguments retrieved in the original Russian language. This result can be highly correlated with the quality of the machine translation system.

4.4 Measuring Argument Similarity

To understand whether the arguments retrieved by the two systems are lexically and semantically similar and whether there is a need to replicate the whole system instead of translating, we calculate similarity scores between arguments retrieved by two systems for 78 comparison objects (from 39 pairs).

The scores are reported in Table 4. To calculate ROUGE-1 and ROUGE-2, we tokenize and lemmatize the arguments since Russian is a highly inflected language. The resulting ROUGE scores are relatively low (ROUGE-1 is 0.257 and ROUGE-2 is 0.046), indicating that the translated texts are lexically quite dissimilar to the texts retrieved by the RuCAM system.

To calculate cosine similarity scores, we use the three following sentence embedding models: (1) multilingual LaBSE [14],[11] (2) Russian BERT large uncased,[12] and (3) Sentence RuBERT,[13] where sentence representations are mean-pooled token embeddings analogous Sentence-BERT [32]. The highest mean cosine similarity score between different embeddings is 0.656, which is borderline, however, similarities might dramatically differ for the arguments from different pairs. For example, the arguments for the pairs "Chinese vs. Western medicine" have a mean similarity of 0.763, and for the "morning vs. afternoon sun", the mean score is 0.290 (examples for these two cases are in Tables 5 and 6).

We also looked at how pairs are ranked according to the ROUGE-1 and cosine similarity metrics of their arguments. First of all, our goal was to check to what extent two metrics are related when identifying similar arguments from two systems. We measured the Spearman's correlation coefficient between ROUGE-1 and cosine similarity, which showed a weak correlation of 0.371 (p-value = 0.02).

[11] https://huggingface.co/sentence-transformers/LaBSE.

[12] https://huggingface.co/ai-forever/sbert_large_nlu_ru.

[13] https://huggingface.co/DeepPavlov/rubert-base-cased-sentence.

Table 5. Example arguments for the object pair 'Chinese medicine vs. Western medicine' that has the highest mean cosine similarity 0.763 among all the object pairs ('rubert-base-cased-sentence' embeddings). For the demonstration purpose, we translated Russian arguments into English using OPUS-MT.

Translation-based	Replication-based (RuCAM)
The amazing thing is that with Traditional Chinese Medicine I always get better faster than all of my colleagues who are relying on Western medicine	"I think more and more Western doctors are realizing today that Chinese medicine is effective," says Dr. Li
Chinese medicine is superior to Western medicine	Chinese medicine has outstripped Western medicine in some respects
As for the treatment of Nephrotic syndrome, by large, Chinese medicine is superior to Western medicine	In Chinese medicine, attention is paid to hidden factors, whereas Western medicine pays more attention to visible indicators
What I am saying is Chinese medicine is a better method of healthcare than Western medicine	In Chinese medicine, for example, kidneys are given much more attention than in Western medicine
I am a firm believer that Chinese medicine is better than Western in many cases	Chinese medicine has coped with what European medicine has not coped with

From Table 7, one can also see that pairs with the most similar and dissimilar arguments do not overlap much, especially between the top-10 object pairs: 'Chinese medicine vs. Western medicine', 'steel knives vs. ceramic knives', and 'Google vs. Yandex search'. According to cosine similarity, more general or common knowledge concepts get higher scores, while for the ROUGE-1 metric, top-10 similar arguments are for companies, brands, and specific topics like programming or medicine. Surprisingly, 'kids vs. adults', 'rain water vs. tap water', and 'skiing vs. snowboarding' appear at the top of cosine similarity scores but at the bottom of the list for the ROUGE-1 score. 'BMW vs. Audi', 'Kenya vs. Tanzania', and 'morning sun vs. afternoon sun' are object pairs that were shown to be different by both metrics. Secondly, our goal was to see, how similar were the arguments from two systems regarding both metrics. Manual analysis of the pairs that were scored differently by ROUGE-1 and cosine similarity showed that high ROUGE-1 scores indeed represent similar arguments, while low cosine similarity scores for those cases can be explained by the unequal number of arguments for each language that increases the impact of the outliers.

Thus, we conclude that a good approach for extending CAM is to combine the translation- and replica-based systems: the results show that the lexical similarity of the arguments from both systems is quite low, while the similarity according to semantic representations is borderline. We also analyzed the arguments to understand whether the dissimilarities could be explained by cultural differences present in the source languages. We identified the following main trends:

Table 6. Example arguments for the object pair 'morning sun vs. afternoon sun' that has the lowest mean cosine similarity 0.290 among all the object pairs ('rubert-base-cased-sentence' embeddings). For the demonstration purpose, we translated Russian arguments into English using OPUS-MT.

Translation-based	Replication-based (RuCAM)
And remember: morning sun is cooler than afternoon sun	Gerberas can be grown in full sun, but it is better in the morning sun and in the midday shade
The morning sun is cooler and gentler than the afternoon hot sun	The location is sunny, but the bright afternoon sun is less useful, shaded
Morning sun is better than afternoon sun	The morning sun is best with reflected light the rest of the time
Early morning sun is better than late afternoon sun since the flowers last longer under cooler conditions	Hot summer sun Many rhododendrons tolerate the morning sun better, although there are some species and varieties that do not tolerate the sun at all
Experienced gardeners know it, morning sun is cooler than afternoon sun.	The morning sun is always preferable to the midday sun, which can burn plants

(a) The retrieved arguments address different aspects of the culture and everyday life of the source language speakers. For example, when comparing car brands, English arguments tend to care more about *safety*, *engine capacities*, and *technology*, while Russian arguments pay attention to *price*, *repair costs*, *wear and tear*, and car *modifications* present on the Russian car market.

(b) Cultural bias occurs in both more specific and more generic comparisons. For instance, for the 'IELTS vs. TOEFL' object pair (more specific comparison), English arguments focus on *complexity* and the test's *specific features*, whereas Russian arguments mainly discuss the *certificate's recognition in other countries.* For the 'skiing vs. snowboarding' pair (more generic), English arguments discuss the *learning rate* and *complexity*, whereas Russian arguments care more about *adrenaline*, *safety* and which sport is *better for families.*

(c) However, for some more generic comparisons like 'football vs. basketball' or 'Western medicine vs. Chinese medicine', the arguments mostly compare the same aspects like *effectiveness*, *popularity*, and often express *personal preferences.*

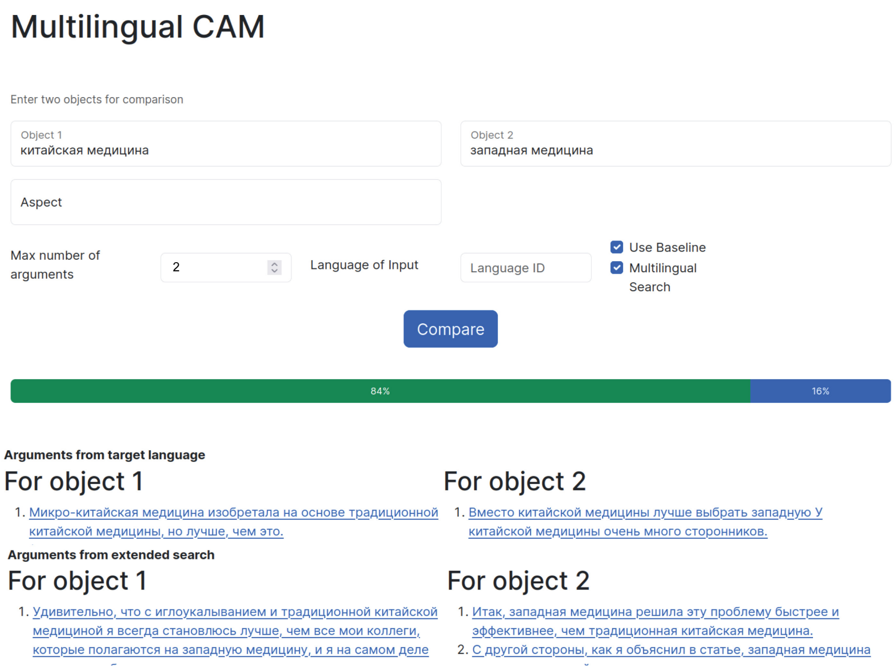
The aforementioned examples highlight that the provenance of retrieved arguments (the language in particular) significantly influences their diversity and introduces potential cultural nuances. In the process of adapting the CAM system to a new language, meticulous consideration should be given to various facets, including the translation quality, the cultural predisposition inherent

Table 7. Object pairs and similarity scores between the retrieved arguments by two systems: translation-based and replica-based. The object pairs are sorted in descending order of the similarity scores. Highlighted in green are the pairs that get high/low/medium scores by the two similarity metrics. Highlighted in red are the pairs showing discrepancies in two similarity metrics.

Cosine Similarity		ROUGE-1	
Chinese vs. Western medicine	0.763	cow milk vs. goat milk	0.393
Apple vs. Google	0.744	rain water vs. tap water	0.362
PHP vs. Python	0.736	London vs. Paris	0.335
Linux vs. Windows	0.732	Chinese vs. Western medicine	0.330
artificial sweeteners vs. white sugar	0.728	skiing vs. snowboarding	0.328
steel knives vs. ceramic knives	0.721	kids vs. adults	0.321
Ibuprofen vs. Aspirin	0.718	steel knives vs. ceramic knives	0.307
hybrid vs. diesel	0.701	Google vs. Yahoo search	0.302
Google vs. Yahoo search	0.700	train vs. plane	0.301
OpenGL vs. Direct3D	0.687	Internet Explorer vs. Firefox	0.292
ASP vs. PHP	0.677	artificial sweeteners vs. white sugar	0.287
NetBeans vs. Eclipse	0.674	basketball vs. football	0.287
Xbox vs. PlayStation	0.669	cats vs. dogs	0.286
laptop vs. desktop	0.661	IELTS vs. TOEFL	0.284
Canon vs. Nikon	0.650	Apple vs. Google	0.284
electric stove vs. gas stove	0.645	electric stove vs. gas stove	0.270
IELTS vs. TOEFL	0.643	Ibuprofen vs. Aspirin	0.265
cow milk vs. goat milk	0.626	OpenGL vs. Direct3D	0.251
quicksort vs. merge sort	0.621	gas vs. charcoal	0.249
Family Guy vs. The Simpsons	0.619	Xbox vs. PlayStation	0.249
basketball vs. football	0.608	Linux vs. Windows	0.246
MAC vs. PC	0.607	pasta vs. pizza	0.246
Adidas vs. Nike	0.596	ASP vs. PHP	0.235
Ford vs. Toyota	0.592	hybrid vs. diesel	0.229
gas vs. charcoal	0.592	laptop vs. desktop	0.224
train vs. plane	0.591	PHP vs. Python	0.223
London vs. Paris	0.590	NetBeans vs. Eclipse	0.211
pasta vs. pizza	0.586	Python vs. R	0.211
Pepsi vs. Coca-cola	0.585	Boeing vs. Airbus	0.211
Internet Explorer vs. Firefox	0.583	MAC vs. PC	0.208
cats vs. dogs	0.581	Family Guy vs. The Simpsons	0.203
Boeing vs. Airbus	0.573	quicksort vs. merge sort	0.188
kids vs. adults	0.567	Ford vs. Toyota	0.185
Python vs. R	0.561	Adidas vs. Nike	0.181
rain water vs. tap water	0.560	Canon vs. Nikon	0.175
skiing vs. snowboarding	0.559	morning sun vs. afternoon sun	0.168
BMW vs. Audi	0.528	BMW vs. Audi	0.154
Kenya vs. Tanzania	0.305	Pepsi vs. Coca-cola	0.151
morning sun vs. afternoon sun	0.290	Kenya vs. Tanzania	0.127

in the source language, and the preferences of the target users—whether they seek responses tailored to a specific language and culture or a more expansive overview. However, in general, a recommended strategy is to merge the outputs of translated arguments and arguments in the target language, thereby enhancing the topical coverage.

5 Multilingual CAM



Multilingual CAM

Enter two objects for comparison

Object 1: китайская медицина

Object 2: западная медицина

Aspect

Max number of arguments: 2

Language of Input: Language ID

☑ Use Baseline
☑ Multilingual Search

Arguments from target language

For object 1

1. Микро-китайская медицина изобретала на основе традиционной китайской медицины, но лучше, чем это.

For object 2

1. Вместо китайской медицины лучше выбрать западную У китайской медицины очень много сторонников.

Arguments from extended search

For object 1

1. Удивительно, что с иглоукалыванием и традиционной китайской медициной я всегда становлюсь лучше, чем все мои коллеги, которые полагаются на западную медицину, и я на самом деле

For object 2

1. Итак, западная медицина решила эту проблему быстрее и эффективнее, чем традиционная китайская медицина.
2. С другой стороны, как я объяснил в статье, западная медицина

Fig. 2. The multilingual CAM interface. Shown are the results for a comparison 'Chinese medicine vs. Western medicine' in Russian. The output combines arguments in the target language (upper part) and translated from English (lower part). The 'Chinese medicine'-object (left-hand) "wins" a comparison (see the bar in the middle).

To showcase the combined approach, we develop a demonstration of multilingual CAM that allows to search for arguments in English CAM and Russian RuCAM via their respective APIs. The interface of a combined system is shown in Fig. 2. It accepts a pair of comparison objects and an optional comparison aspect in either language and retrieves arguments in both languages when the option 'Multilingual Search' is selected. Otherwise, the answers are provided in the input languages. Optionally, the user can specify the input language (e.g., when searching for “BMW” written in Latin script in the Russian texts); otherwise, the language is identified automatically. The output arguments are grouped into two blocks: those that come from the corpus of an input language and (in the case of the multilingual search option) translated from another language.

Our first prototype of multilingual CAM currently has several technical limitations. First, it depends on the successful response from the CAM or RuCAM API. Second, it relies on the machine translation module, which may incorrectly translate the user input, resulting in a failure to find relevant arguments. Therefore, future work should focus on overcoming the aforementioned shortcomings by locally hosting the retrieval corpora and deploying other translation models.

6 Conclusion

In this paper, we improved the answer stance detection of CAM and RuCAM—systems that can answer comparative questions in English and Russian—by fine-tuning RoBERTa-based models. Furthermore, we compared the replica-based RuCAM approach of “localizing” CAM to the Russian language to a simple machine translation-based CAM variant. Our analyses showed that translating CAM’s inputs and outputs also yields decent effectiveness scores with respect to result relevance and quality. However, we also found that the results retrieved by the two systems (translation-based and replica-based) are lexically and semantically quite dissimilar as, for instance, the Russian results from the replica-based RuCAM system that uses native Russian data can be more culture-specific and might take into account uncommon and unexpected aspects. Therefore, combining the results of the translation-based and of the replica-based CAM variants could yield more diverse arguments for comparisons.

As a demonstrator of a multilingual CAM system, we implemented an interface to combine the results of translation- and replica-based CAM systems. In a user study, we found that, for instance, the perceived result quality is highly dependent on the translation quality; in our study, translated results were perceived as more relevant but of a lower quality than the results retrieved in the target language—often also related to the translation quality of the actual search terms (comparison objects and aspects).

Limitations

Our current work focused on two rather high-resource languages (English and Russian) so that our findings and conclusions may not be applicable to lower-resource languages. In future research, we thus plan to also analyze CAM-like systems in other languages.

Furthermore, our study results depend on two restricting factors: (1) the choice of the machine translation model, and (2) potential biases of the manual annotations. To alleviate the first factor, we relied on previous work and preferred publicly available translation models that can be easily deployed. As for the second factor, while annotation bias cannot be fully avoided, we ensured that our annotators understood and followed the guidelines by conducting pilot annotations with a follow-up discussion of possible misinterpretations. In the future, larger studies with a bigger group of human annotators are necessary for more robust conclusions.

Finally, CAM and RuCAM operate on large document collections, in which the amount of relevant data cannot be controlled or measured. To more closely study sociocultural questions in the context of comparison analyses, other more focussed collections might be better suited.

Acknowledgements. This work has been partially supported by the DFG (German Research Foundation) through the project "ACQuA 2.0: Answering Comparative Questions with Arguments" (project 376430233) as part of the priority program "RATIO: Robust Argumentation Machines" (SPP 1999).

References

1. Allaway, E., McKeown, K.R.: Zero-shot stance detection: a dataset and model using generalized topic representations. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, pp. 8913–8931. ACL (2020). https://doi.org/10.18653/V1/2020.EMNLP-MAIN.717
2. Anand, P., Walker, M.A., Abbott, R., Tree, J.E.F., Bowmani, R., Minor, M.: Cats rule and dogs drool! Classifying stance in online debate. In: Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis, WASSA@ACL 2011, pp. 1–9. ACL (2011). https://aclanthology.org/W11-1701/
3. Bar-Haim, R., Bhattacharya, I., Dinuzzo, F., Saha, A., Slonim, N.: Stance classification of context-dependent claims. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, pp. 251–261. ACL (2017). https://doi.org/10.18653/V1/E17-1024
4. Bondarenko, A., Ajjour, Y., Dittmar, V., Homann, N., Braslavski, P., Hagen, M.: Towards understanding and answering comparative questions. In: Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, WSDM 2022, pp. 66–74. ACM (2022). https://doi.org/10.1145/3488560.3498534
5. Bondarenko, A., et al.: Overview of Touché 2020: argument retrieval. In: Arampatzis, A., et al. (eds.) CLEF 2020. LNCS, vol. 12260, pp. 384–395. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58219-7_26
6. Bondarenko, A., et al.: Overview of Touché 2022: argument retrieval. In: Barrón-Cedeño, A., et al. (eds.) CLEF 2022. LNCS, vol. 13390, pp. 311–336. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-13643-6_21
7. Bondarenko, A., et al.: Overview of Touché 2021: argument retrieval. In: Candan, K.S., et al. (eds.) CLEF 2021. LNCS, vol. 12880, pp. 450–467. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-85251-1_28
8. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2016, pp. 785–794. ACM (2016). https://doi.org/10.1145/2939672.2939785
9. Conneau, A., et al.: Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, pp. 8440–8451. ACL (2020). https://doi.org/10.18653/V1/2020.ACL-MAIN.747

10. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, pp. 670–680. Association for Computational Linguistics (2017). https://doi.org/10.18653/V1/D17-1070
11. Cortes, C., Vapnik, V.: Support-vector networks. Mach. Learn. **20**(3), 273–297 (1995). https://doi.org/10.1007/BF00994018
12. Cox, D.R.: The regression analysis of binary sequences. J. Roy. Stat. Soc. Ser. B (Methodol.) **20**(2), 215–232 (1958)
13. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, pp. 4171–4186. ACL (2019). https://doi.org/10.18653/V1/N19-1423
14. Feng, F., Yang, Y., Cer, D., Arivazhagan, N., Wang, W.: Language-agnostic BERT sentence embedding. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, ACL 2022, pp. 878–891. ACL (2022). https://doi.org/10.18653/V1/2022.ACL-LONG.62
15. Hasan, K.S., Ng, V.: Stance classification of ideological debates: data, models, features, and constraints. In: Proceedings of the Sixth International Joint Conference on Natural Language Processing, IJCNLP 2013, pp. 1348–1356. AFNLP/ACL (2013). https://aclanthology.org/I13-1191/
16. Igarashi, Y., Komatsu, H., Kobayashi, S., Okazaki, N., Inui, K.: Tohoku at SemEval-2016 task 6: feature-based model versus convolutional neural network for stance detection. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, pp. 401–407. ACL (2016). https://doi.org/10.18653/V1/S16-1065
17. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of IR techniques. ACM Trans. Inf. Syst. **20**(4), 422–446 (2002). https://doi.org/10.1145/582415.582418
18. Kang, I., et al.: LLM-augmented preference learning from natural language. arXiv 2310.08523 (2023). https://doi.org/10.48550/arXiv.2310.08523
19. Küçük, D., Can, F.: Stance detection: a survey. ACM Comput. Surv. **53**(1), 12:1–12:37 (2021). https://doi.org/10.1145/3369026
20. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach. arXiv 1907.11692 (2019). http://arxiv.org/abs/1907.11692
21. Ma, N., Mazumder, S., Wang, H., Liu, B.: Entity-aware dependency-based deep graph attention network for comparative preference classification. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, pp. 5782–5788. ACL (2020). https://doi.org/10.18653/v1/2020.acl-main.512
22. Maslova, M., Rebrikov, S., Artsishevski, A., Zaczek, S., Biemann, C., Nikishina, I.: RuCAM: comparative argumentative machine for the Russian language. In: Ignatov, D.I., et al. (eds.) AIST 2023. LNCS, vol. 14486, pp. 78–91. Springer, Cham (2024). https://link.springer.com/chapter/10.1007/978-3-031-54534-4_6
23. Mohammad, S.M., Kiritchenko, S., Sobhani, P., Zhu, X., Cherry, C.: SemEval-2016 task 6: detecting stance in tweets. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, pp. 31–41. ACL (2016). https://doi.org/10.18653/V1/S16-1003
24. Murakami, A., Raymond, R.: Support or oppose? Classifying positions in online debates from reply activities and opinion expressions. In: Proceedings of the 23rd

International Conference on Computational Linguistics, COLING 2010, pp. 869–875. CIPS (2010). https://aclanthology.org/C10-2100/

25. Nadamoto, A., Tanaka, K.: A comparative web browser (CWB) for browsing and comparing web pages. In: Proceedings of the Twelfth International World Wide Web Conference, WWW 2003, pp. 727–735. ACM (2003). https://doi.org/10.1145/775152.775254
26. OpenAI: Introducing ChatGPT (2022). https://openai.com/blog/chatgpt
27. OpenAI: GPT-4 technical report. ArXiv 2303.08774 (2023). https://doi.org/10.48550/arXiv.2303.08774
28. Ortiz Suárez, P.J., Sagot, B., Romary, L.: Asynchronous pipelines for processing huge corpora on medium to low resource infrastructures. In: Proceedings of the Workshop on Challenges in the Management of Large Corpora, CMLC-7 2019, pp. 9–16. IDS (2019). https://doi.org/10.14618/ids-pub-9021
29. Panchenko, A., Bondarenko, A., Franzek, M., Hagen, M., Biemann, C.: Categorizing comparative sentences. In: Proceedings of the 6th Workshop on Argument Mining, ArgMining@ACL 2019, pp. 136–145. ACL (2019). https://doi.org/10.18653/V1/W19-4516
30. Panchenko, A., Ruppert, E., Faralli, S., Ponzetto, S.P., Biemann, C.: Building a web-scale dependency-parsed corpus from common crawl. In: Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018. ELRA (2018). http://www.lrec-conf.org/proceedings/lrec2018/summaries/215.html
31. Rajadesingan, A., Liu, H.: Identifying users with opposing opinions in twitter debates. In: Kennedy, W.G., Agarwal, N., Yang, S.J. (eds.) SBP 2014. LNCS, vol. 8393, pp. 153–160. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-05579-4_19
32. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using Siamese BERT-networks. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, pp. 3980–3990. ACL (2019). https://doi.org/10.18653/V1/D19-1410
33. Schildwächter, M., Bondarenko, A., Zenker, J., Hagen, M., Biemann, C., Panchenko, A.: Answering comparative questions: better than ten-blue-links? In: Proceedings of the 2019 Conference on Human Information Interaction and Retrieval, CHIIR 2019, pp. 361–365. ACM (2019). https://doi.org/10.1145/3295750.3298916
34. Schiller, B., Daxenberger, J., Gurevych, I.: Stance detection benchmark: how robust is your stance detection? Künstliche Intell. **35**(3), 329–341 (2021). https://doi.org/10.1007/S13218-021-00714-W
35. Sun, J., Wang, X., Shen, D., Zeng, H., Chen, Z.: CWS: a comparative web search system. In: Proceedings of the 15th International Conference on World Wide Web, WWW 2006, pp. 467–476. ACM (2006). https://doi.org/10.1145/1135777.1135846
36. Thomas, M., Pang, B., Lee, L.: Get out the vote: determining support or opposition from congressional floor-debate transcripts. In: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, EMNLP 2006, pp. 327–335. ACL (2006). https://aclanthology.org/W06-1639/
37. Tiedemann, J., Thottingal, S.: OPUS-MT - building open translation services for the world. In: Proceedings of the 22nd Annual Conference of the European Association for Machine Translation, EAMT 2020, pp. 479–480. European Association for Machine Translation (2020). https://aclanthology.org/2020.eamt-1.61/

38. Touvron, H., et al.: LLaMA: open and efficient foundation language models. arXiv 2302.13971 (2023). https://doi.org/10.48550/arXiv.2302.13971
39. Vamvas, J., Sennrich, R.: X-stance: a multilingual multi-target dataset for stance detection. In: Proceedings of the 5th Swiss Text Analytics Conference and the 16th Conference on Natural Language Processing, SwissText/KONVENS 2020. CEUR Workshop Proceedings, vol. 2624. CEUR-WS.org (2020). https://ceur-ws.org/Vol-2624/paper9.pdf
40. Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: Proceedings of the 6th International Conference on Learning Representations, ICLR 2018. OpenReview.net (2018). https://openreview.net/forum?id=rJXMpikCZ
41. Walker, M.A., Anand, P., Abbott, R., Grant, R.: Stance classification using dialogic properties of persuasion. In: Proceedings of the Conference of the North American Chapter of the Association of Computational Linguistics, NAACL-HLT 2012, pp. 592–596. ACL (2012). https://aclanthology.org/N12-1072/
42. Walker, M.A., Anand, P., Abbott, R., Tree, J.E.F., Martell, C.H., King, J.: That is your evidence? Classifying stance in online political debate. Decis. Support Syst. **53**(4), 719–729 (2012). https://doi.org/10.1016/J.DSS.2012.05.032
43. Wei, W., Zhang, X., Liu, X., Chen, W., Wang, T.: pkudblab at SemEval-2016 task 6: a specific convolutional neural network system for effective stance detection. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, pp. 384–388. ACL (2016). https://doi.org/10.18653/V1/S16-1062
44. Yu, N., Pan, D., Zhang, M., Fu, G.: Stance detection in Chinese MicroBlogs with neural networks. In: Lin, C.-Y., Xue, N., Zhao, D., Huang, X., Feng, Y. (eds.) ICCPOL/NLPCC -2016. LNCS (LNAI), vol. 10102, pp. 893–900. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-50496-4_83
45. Zarrella, G., Marsh, A.: MITRE at SemEval-2016 task 6: transfer learning for stance detection. In: Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, pp. 458–463. ACL (2016). https://doi.org/10.18653/V1/S16-1074

Objective Argument Summarization in Search

Timon Ziegenbein[1(✉)], Shahbaz Syed[2], Martin Potthast[3],
and Henning Wachsmuth[1]

[1] Leibniz University Hannover, Hanover, Germany
{t.ziegenbein,h.wachsmuth}@ai.uni-hannover.de
[2] Leipzig University, Leipzig, Germany
shahbaz.syed@uni-leipzig.de
[3] hessian.AI, and ScaDS.AI, University of Kassel, Kassel, Germany
martin.potthast@uni-kassel.de

Abstract. Decision-making and opinion formation are influenced by arguments from various online sources, including social media, web publishers, and, not least, the search engines used to retrieve them. However, many, if not most, arguments on the web are informal, especially in online discussions or on personal pages. They can be long and unstructured, subjective and emotional, and contain inappropriate language. This makes it difficult to find relevant arguments efficiently. We hypothesize that, on search engine results pages, "objective snippets" of arguments are better suited than the commonly used extractive snippets and develop corresponding methods for two important tasks: *snippet generation* and *neutralization*. For each of these tasks, we investigate two approaches based on (1) prompt engineering for large language models (LLMs), and (2) supervised models trained on existing datasets. We find that a supervised summarization model outperforms zero-shot summarization with LLMs for snippet generation. For neutralization, using reinforcement learning to align an LLM with human preferences for suitable arguments leads to the best results. Both tasks are complementary, and their combination leads to the best snippets of arguments according to automatic and human evaluation.

Keywords: Computational Argumentation · Information Retrieval · Large Language Models · Text Summarization · Text Neutralization

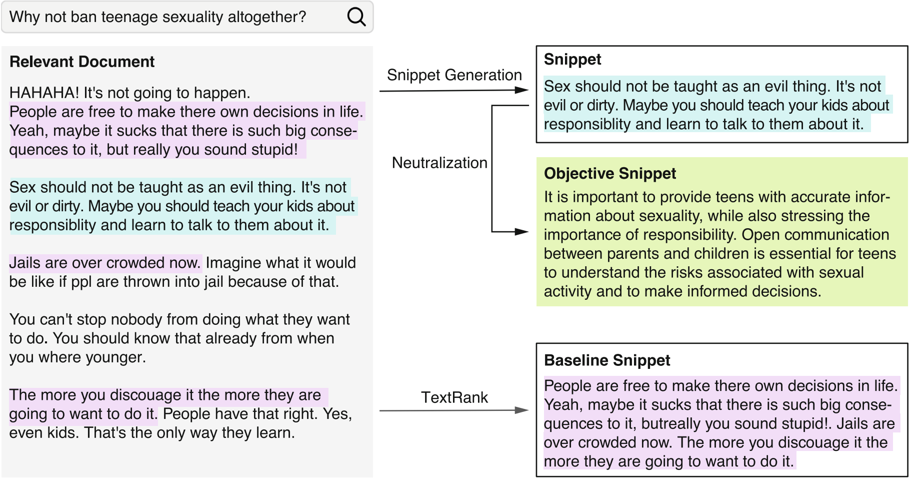
1 Introduction

Deliberative processes are a key element of well-informed decision-making and opinion formation. Their goal is to explore and evaluate the space of arguments that are relevant for deciding on the best course of action in a given situation [34]. Vast amounts of arguments on virtually all topics of interest can be

T. Ziegenbein and S. Syed—Equal contribution.

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 335–351, 2024.
https://doi.org/10.1007/978-3-031-63536-6_20

found on the web and are retrievable using generic or specialized search engines. However, the argument snippets returned by argument search engines are often insufficient to help users find relevant arguments—for two main reasons. First, the standard methods for generating snippets often fail to capture the essence of an argument [2] (henceforth referred to as the argument's "gist"). Second, the snippets often contain subjective, informal, emotional, or inappropriate language that distracts from the gist [38]. Though the original arguments may still contain information that is highly relevant to a topic, snippets that reflect inappropriate presentations may prevent users from recognizing them as relevant.



Relevant Document

HAHAHA! It's not going to happen. People are free to make there own decisions in life. Yeah, maybe it sucks that there is such big consequences to it, but really you sound stupid!

Sex should not be taught as an evil thing. It's not evil or dirty. Maybe you should teach your kids about responsiblity and learn to talk to them about it.

Jails are over crowded now. Imagine what it would be like if ppl are thrown into jail because of that.

You can't stop nobody from doing what they want to do. You should know that already from when you where younger.

The more you discouage it the more they are going to want to do it. People have that right. Yes, even kids. That's the only way they learn.

Snippet

Sex should not be taught as an evil thing. It's not evil or dirty. Maybe you should teach your kids about responsiblity and learn to talk to them about it.

Objective Snippet

It is important to provide teens with accurate information about sexuality, while also stressing the importance of responsibility. Open communication between parents and children is essential for teens to understand the risks associated with sexual activity and to make informed decisions.

Baseline Snippet

People are free to make there own decisions in life. Yeah, maybe it sucks that there is such big consequences to it, butreally you sound stupid!. Jails are over crowded now. The more you discouage it the more they are going to want to do it.

Fig. 1. Illustration of our two-step approach encompassing snippet generation and neutralization to create an objective snippet of a relevant document (argumentative text) for a user query (controversial issue). The document contains information that is relevant to the query, although written inappropriately. Our objective snippet mitigates this while retaining the relevant content. For comparison, an extractive TextRank baseline reflects this inappropriateness, resulting in a potentially ineffective snippet.

In this paper, we investigate whether "objective snippets" are better suited for argument search engines. We define such a snippet to combine the main claim of an argument and the evidence supporting it (basically, the gist), while avoiding overly subjective and informal language. We propose a two-step approach to create objective snippets of arguments. The first step, snippet generation, aims to extract the main message and supporting evidence of an argument. We assume that a short summary of an argument (i.e. two sentences) can represent this gist. The second step, neutralization, aims to neutralize the language of the extracted core statement to make it more objective. We also investigate the necessity of neutralization as a separate task, since abstractive summaries in particular can potentially neutralize the language of the source text already during generation. Figure 1 exemplifies snippets from existing snippet generation models as well as from our approach. It demonstrates that existing approaches produce snippets that retain inappropriate language, which undermines the effectiveness of the main argument. In contrast, our approach combines snippet generation and neutralization to produce objective, well-written snippets while preserving the semantics of the source argument. Our contributions are as follows:[1]

- A two-step approach that tackles the tasks of snippet generation and neutralization to create objective argument snippets for argument search (Sect. 3).
- Three manual evaluation studies on snippet generation and neutralization, individually and in combination, using (1) the args.me corpus [1] and (2) the appropriateness corpus [38] as ground truth (Sects. 4 and 5).

We show that abstractive snippets are better suited to present arguments as search results than extractive snippets. In particular, argument neutralization leads to an expected increase in the likelihood of a productive discussion on the topic. Moreover, combining abstractive summarization with neutralization creates a more objective snippet that further improves the already-preferred abstractive snippets in terms of the likelihood that users are willing to read the full argument presented by the snippet.

2 Related Work

In this section, we describe relevant previous work on the tasks of snippet generation and neutralization. Since snippet generation is very similar to summarization, we describe relevant work from both areas.

2.1 Snippet Generation

Snippets in search engines are primarily extractive in nature. Snippet generators extract the most relevant parts of the source text, especially those containing the terms of the query [3,14,32,35]. The aim of a snippet is to help users quickly identify documents likely to satisfy their information need [9]. First, argument search engines such as args.me [33] or ArgumenText [28] used the first sentences of retrieved arguments as snippets. Later, extractive snippets of the arguments as proposed by Alshomary et al. [2] replaced them, enriching TextRank with argumentative information to extract the main claim and supporting premise as an argument snippet, which forms a baseline in our evaluation. The arguments were also summarized in individual sentences [28], key points [4], and conclusions [30].

Our motivation is to introduce objective snippets of arguments in a search engine. While minimizing the reuse of text in the snippets (from the source) is beneficial [7], traditionally, extractive summaries are preferred over abstractive summaries to avoid incorrect rephrasing of facts from the source text. This is because abstractive summaries of standard sequence-to-sequence models suffer

[1] The experiment code is available at https://github.com/webis-de/RATIO-24.

from hallucinations [24] and incorrectly merge different parts of the source, leading to incorrect facts [5]. However, recent advances in abstractive summarization using pre-trained language models have been shown to generate more fluid and coherent summaries than purely extractive approaches, which improves their overall readability and preference by humans [13]. Therefore, we opt for abstractive snippets in this work. Moreover, we investigate the zero-shot effectiveness of the instruction-driven Alpaca [31] model using prompting.

2.2 Neutralization

Neutralization can be seen as a style transfer task. Style transfer in the context of natural language generation aims to control attributes in the generated text, such as politeness, emotion, or humor among many others [17]. Text style transfer has been applied to authorial features and literary genres [12]. Most studies deal with broad notions of style, including the formality and subjectivity of a text [18]. There are also approaches to changing sentiment polarity (of reviews) [16], political bias (of news headlines) [6], and framing (of news articles) [8].

Many approaches learn a sequence-to-sequence model on parallel source–target text pairs. Modifying the style often works reliably, but preserving the content seems to be a challenge [6]. On the other hand, style and content are difficult to separate in text (i.e., words can reflect both simultaneously). To mitigate this, some works avoid disentangling latent representations of style and content [10], but this cannot guarantee that certain information is preserved. Others restrict transfer to low-level linguistic decisions [12,27].

Our aim is to improve the appropriateness of arguments to ensure that they are suitable for a wide audience. However, unlike traditional style transfer, the role of semantic preservation here is rather superficial, as some parts of our texts that are responsible for inappropriateness may be inappropriate due to their content rather than their style, such as ad hominem attacks. Therefore, we generally prefer appropriateness over semantic similarity in this paper.[2] Since no parallel data is available for the argument neutralization task, we rely on an instruction-based zero-shot approach with Alpaca [31]. For further refinement, we use the appropriateness classifier from Ziegenbein et al. [38] and an adapted version of the RLHF (Reinforcement Learning using Human Feedback) method from Stiennon et al. [29]. The authors of Madanagopal and Caverlee [23] use a reinforcement learning-based approach to correct subjective language in Wikipedia articles, which comes closest to our approach. However, their approach is based on parallel data, which is not available for the task of neutralization. As far as we know, there is no style transfer approach for argument neutralization to date, and none of the related reinforcement learning approaches for style transfer use prompting as the initial model (i.e., for the policy).

[2] The role of semantic similarity is being investigated in another paper under review.

3 Approach

This section describes the approaches we evaluated for generating argument snippets and their neutralization.

3.1 Snippet Generation

We investigated three snippet generation approaches: (1) an unsupervised extractive argument summarization model, (2) a supervised abstractive news summarization model, and (3) an instruction-tuned zero-shot summarization model.

Extractive-Summarizer. With TextRank, Alshomary et al. [2] proposed an unsupervised extractive argument snippet generation approach that extracts the main claim and premise of an argument as its snippet. To identify the corresponding argument sentences, a variant of PageRank [26] is used to rank them based on their contextual importance and argumentativeness. Starting from equal scores for all sentences, the model iteratively updates these scores until convergence is achieved. The two highest-scoring sentences are then extracted in their original order to maintain coherence. TextRank serves as the standard model for generating snippets for the args.me search engine and as our baseline.

Abstractive-Summarizer. For supervised snippet generation, we use a BART model [21], finetuned to the task of abstractive news summarization on the CNN/DailyMail dataset [25].[3] To tailor its summaries to the task argument snippet generation, we shorten the input to 102 tokens and limit the minimum and maximum summary length to 25% and 35% of the argument length respectively.

Instruction-Summarizer. To instruct Alpaca to generate a snippet, we use the prompt `### Instruction: The following is an argument on the topic"<topic>". Extract a coherent gist from it that is exactly two sentences long. ### Input: <argument> ### Response:` and insert an argument and its topic. Generation is done at a temperature of 1 and sampling with a p-value of 0.95. The number of generated sentences is limited to two in order to ensure snippets of a similar length compared to the other approaches.

3.2 Neutralization

For neutralization, we compare (1) an instruction-tuned zero-shot neutralization model, and (2) a reinforcement learning-aligned neutralization model.

Instruction-Neutralizer. To instruct Alpaca to neutralize a text, we use the prompt `### Instruction: Rewrite the following argument on the topic of "<topic>" to be more appropriate and make only minimal changes to the original argument. ### Input: <argument> ### Response:` and provide it with the argument and its topic. We use a temperature of 1 and sample with a p-value of 0.95 during generation. The number of generated tokens is limited

[3] https://huggingface.co/facebook/bart-large-cnn.

to 50% to 150% of the original argument to ensure that the model does not delete or add too much content when rewriting the arguments or snippets.

Aligned-Neutralizer. To align Alpaca with human-defined appropriateness criteria, we finetune it using reinforcement learning from human feedback [29,39]. During the training process, we use the same prompt settings and hyperparameters as before, but adjust the output of the model to generate texts that are categorized as appropriate by the appropriateness classifier of Ziegenbein et al. [38]. Thus, texts generated by Alpaca serve as input to the classifier and the returned probability value for the appropriateness class as a reward to update Alpaca. For efficiency, we do not update Alpaca's original weights but use adapter-based low-rank adaptation (LoRA) [15]. A full description of the approach and the training process is part of a paper soon to be published [37].[4]

4 Data

For evaluation, we use two datasets sampled from (1) the args.me corpus [1] and (2) the appropriateness corpus [38]. The former is used to evaluate the snippet generation approaches and combining snippet generation and neutralization, while the latter is used to evaluate the argument neutralization approaches.

4.1 The args.me Corpus

To obtain the dataset for our snippet generation experiments, we sample arguments from the args.me corpus [1]. The args.me corpus contains 387,606 arguments from four debate portals, each annotated with a stance (pro or con) and a topic (e.g., "abortion" or "gay marriage"). Based on the ten most frequently submitted queries to the args.me API [33], we created an initial dataset. To ensure adequate summarization potential for snippet generation and to account for possible input length limitations of the models used in our experiments, we filter the dataset to contain only arguments between 100 and 500 words in length. Furthermore, we use an ensemble classifier based on the five folds of the appropriateness corpus to retain only inappropriate arguments. Finally, we extract the top five pro and top five con arguments for each query based on the args.me ranking obtained from its API. This gives us a final dataset of 99 arguments.[5]

4.2 The Appropriateness Corpus

To obtain the dataset for our neutralization experiments, we sample arguments from the appropriateness corpus [38]. The corpus contains 2,191 arguments

[4] The code and data used to train the models can be found here: https://github.com/webis-de/RATIO-24.

[5] As one of the queries did not contain enough arguments to meet the inappropriateness criteria, one query contains only nine arguments instead of ten.

labeled with the corresponding discussion titles from three genres (reviews, discussion forums, and Q&A forums). Each argument is annotated by three annotators according to a 14-dimensional taxonomy of inappropriateness errors. We filter the corpus to include only arguments that were classified as inappropriate by all three annotators in the original study to ensure that there is a clear need for neutralization. As before, we only retain arguments between 100 and 500 words in length. Finally, we draw a random sample of 100 arguments from the corpus to obtain our final dataset.

Table 1. Evaluation of the snippet generation approaches without neutralization: (a) ROUGE-1 (R1), ROUGE-2 (R2), ROUGE-L (RL), and BERTScore (Sim.), computed between the source argument and the generated snippet, perplexity (PPL) of the generated snippet and percentage of appropriate generated snippets (App.). (b) Absolute counts of ranks assigned by human evaluators to the three approaches and their average.

Approach	(a) Automatic						(b) Manual			
	R1	R2	RL	Sim	PPL↓	App.↑	#1	#2	#3	Avg.↓
Extractive-Sum.	0.29	0.28	0.29	0.25	67.7	0.21	42	126	**327**	2.58
Abstractive-Sum.	**0.40**	**0.38**	**0.38**	**0.35**	50.9	0.31	**274**	149	72	**1.59**
Instruction-Sum.	0.24	0.11	0.16	0.13	**26.5**	**0.58**	179	**220**	96	1.83

5 Evaluation

We evaluate our approaches in a series of experiments, both automatically and manually. For automatic evaluation, we quantify the content preservation of all approaches with ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL) [22] for lexical similarity, and with BERTScore (Sim.) [36] for semantic similarity. Furthermore, we measure the fluency of the generated texts with Perplexity (PPL) and compute the percentage of instances for which an approach was able to change the label from inappropriate to appropriate (based on the ensemble classifier of Ziegenbein et al. [38], see Sect. 3). The manual evaluation is detailed in the corresponding subsections, as the user studies differ for each of the tasks.

5.1 Snippet Generation

Automatic Evaluation. Table 1a shows that, when automatically determining the best summarization model for snippet generation, the Abstractive-Summarizer scores best in terms of content preservation (highest R1, R2, RL, Sim.). Instruction-Summarizer is strongest in fluency (PPL 26.5) and creates appropriate snippets for 58% of inappropriate arguments. The extractive baseline Extractive-Summarizer does not win in any of the automatic measurements used.

Manual Evaluation. We hired five evaluators on upwork.com who are native English speakers and tasked them to evaluate snippets of 99 arguments from our three models: Instruction-Summarizer, Abstractive-Summarizer, and Extractive-Summarizer. Given a topic, a source argument (pro/con) and three snippets, the evaluators rated the suitability of a snippet to be displayed on a search engine results page for the argument by ranking them from "best" to "worst." A detailed annotation guide describing the characteristics of a good snippet, such as high coverage of key information from the original argument and its ability to help users easily identify relevant arguments from a ranking of results.

As shown in Table 1b, Abstractive-Summarizer proved to be the best model for generating snippets according to the evaluators, ranking first in about 56% of the examples (274 out of 495). The agreement between annotators was 0.22, as measured by Kendall's τ rank correlation coefficient [19]. This indicates a positive rank correlation while underlining the subjectivity of the quality ratings.

5.2 Neutralization

Automatic Evaluation. Comparing the Instruction-Neutralizer with the Aligned-Neutralizer, Table 2a shows that there are differences in content preservation and transfer of appropriateness. That is, the Instruction-Neutralizer performs better on R1 (0.79), R2 (0.66), RL (0.73), and Sim. (0.67), whereas the Aligned-Neutralizer performs better on fluency (PPL 18.4) and transfer (App. 0.97), making almost all arguments appropriate (97%). This suggests that there is a trade-off between retaining the content of the argument and improving appropriateness. As mentioned above, we are investigating this effect in another paper that is not yet published at the time of writing. However, a manual inspection of the neutralized arguments and our annotators' comments shows that, despite the rather low content preservation (0.18 for BERTScore), the main meaning of the argument and its reasoning are mostly preserved, but the arguments do not show any lexical similarity to the original argument.

Table 2. Evaluation of the neutralization approaches: (a) ROUGE-1 (R1), ROUGE-2 (R2), ROUGE-L (RL), BERTScore (Sim.), perplexity (PPL) of the neutralized argument, and percentage of successfully neutralized arguments (App.). (b) Absolute counts of ranks assigned by the human evaluators to the three approaches and their average.

Approach	(a) Automatic						(b) Manual			
	R1	R2	RL	Sim	PPL↓	App.↑	#1	#2	#3	Avg.↓
Exact-Copy	**1.00**	**1.00**	**1.00**	**1.00**	66.1	0.00	10	91	**399**	2.78
Instruction-Neut.	0.79	0.66	0.73	0.67	29.5	0.40	67	**345**	88	2.04
Aligned-Neut.	0.41	0.16	0.27	0.18	**18.4**	**0.97**	**423**	64	13	**1.18**

Manual Evaluation. If people prefer neutralized arguments over the baseline arguments that contain inappropriate content, this is evidence that neutralization is useful for the ultimate goal of creating "objective snippets." Accordingly, we evaluated the neutralized arguments of Instruction-Neutralizer and

Aligned-Neutralizer together with the baseline argument. Like above, five human evaluators ranked the three argument variants from “best” to “worst” according to their appropriateness to be presented in a civil debate on a given topic. We used 100 (manually labeled) inappropriate arguments from the appropriateness corpus. The evaluators were provided with a comprehensive guide describing the characteristics of inappropriate arguments and how to identify them [38].

Table 2b shows the results. Neutralized arguments from Aligned-Neutralizer are preferred over others in 84.6% of cases (423 out of 500). This underlines the effectiveness of neutralization and its implicit goal of making arguments more appropriate in public debates. Kendall’s τ for this evaluation was 0.48, indicating a positive correlation between the rankings. Compared to the snippet generation task, the evaluators were able to distinguish more reliably between the quality of inappropriate and appropriate variants of an argument.

5.3 Objective Snippets

Automatic Evaluation. Comparing the two approaches using our automatic measures, Table 3a shows that combining Abstractive-Summarizer with Aligned-Neutralizer further decreases the similarity of the snippet to the original argument (0.35 vs. 0.11), but increases the number of appropriate snippets (0.87 vs. 0.31).

Table 3. Evaluation of the combined approach (snippet generation + neutralization): (a) ROUGE-1 (R1), ROUGE-2 (R2), ROUGE-L (RL), BERTScore (Sim.), perplexity (PPL) of the generated snippet, and percentage of appropriate snippets generated (App.). (b) Absolute and relative count of snippets of one approach being preferred over the other.

Approach	(a) Automatic						(b) Manual	
	R1	R2	RL	Sim	PPL↓	App.↑	Pref.↑	%↑
Abstractive-Sum.	**0.40**	**0.38**	**0.38**	**0.35**	50.9	0.31	57	0.11
+ Aligned-Neut.	0.25	0.10	0.17	0.11	**20.0**	**0.87**	**438**	**0.89**

Manual Evaluation. In addition to evaluating the individual subtasks, we also evaluated the holistic approach by assessing the usefulness of the objective snippets. Specifically, we performed a pairwise comparison between the objective snippets and the non-neutralized snippets. In contrast to evaluating the generation of the snippets, where the original argument was also provided, we only provided the topic to the five human evaluators. Given a self-contained query, they were asked to select the excerpt they were most likely to click on to read the full argument. For this evaluation, we used 100 arguments for 10 topics from the args.me corpus and selected an equal number of pro and con arguments.

Table 3b shows the results. Objective snippets were preferred over non-neutralized snippets in 89% of the cases (438 out of 495). This indicates that neutralization has a positive effect on the likelihood that search engine users will follow the link to read the full argument from which the snippet was extracted.

Krippendorff's α [20] was 0.29, indicating moderate agreement between annotators. Further examples of snippets generated by our best approach (Abstractive-Sum. + Aligned-Neut.) are shown in Table 5 in the Appendix.

Qualitative Analysis. We conducted a manual evaluation of each task, which included the generation and neutralization of snippets as well as the resulting objective snippets generated with our approach. For all tasks, we recruited annotators who are native English speakers, aiming for a balanced representation of male and female annotators. Annotators had the opportunity to provide comments and could also contact us directly if they needed help. No additional questions were asked throughout the annotation tasks, with the exception of a brief review of a small subset of completed annotations to confirm understanding of the task.

Table 4. Quality dimensions for each tasks (snippet generation, neutralization, objective snippets), derived from the comments of annotators in our manual evaluation studies.

Task	Quality Dimensions (Preferred by Annotators)
Snippet Generation	specificity, clarity, positive/inoffensive language, conciseness, self-containment, informativeness, focus on the issue, avoiding personal attacks, structure and coherence, accuracy/correctness
Neutralization	openness, simple language, absence of profanity, facilitating critical evaluation, seriousness, absence of grammatical/orthographic errors, balanced emotions, well-reasoned, structure and coherence, formal language, non-speculative
Objective Snippets	conciseness, simple language, fluency, balanced emotions, includes quotes/evidence/statistics, specificity, coherence

For each example within our three studies, annotators were asked to provide optional feedback in natural language on their ratings and preferences for the results of each study. We manually analyzed nearly 500 comments to identify important quality dimensions for achieving the goal of creating objective snippets. In particular, we derived quality dimensions that have been studied in related areas such as summarization, text generation, and sentiment analysis. Table 4 provides an overview of these dimensions for each task. Examples of comments for the tasks of snippet generation, neutralization, and objective snippets are shown in Tables 6 and 7 in the Appendix, respectively.

Overall, we found that grammaticality and positive language strongly influenced the credibility and acceptability of the argument snippets. Annotators consistently preferred arguments that were free of spelling errors, had correct punctuation, and were well-structured, regardless of their content. Therefore, ensuring grammatical correctness and a well-structured output is crucial. Furthermore, the use of positive language is preferred over negative language, with annotators emphasizing that a positive tone signals critical thinking and openness to other opinions. Consequently, neutralization plays a key role in ensuring that the snippets are suitable for a wide audience. In line with the quality dimensions of summaries [11], high-quality annotators preferred snippets that were informative, concise and coherent.

Limitations and Ethical Concerns

This paper aims to provide evidence that objective argument snippets significantly improve the overall user experience when searching for arguments. While our human annotators strongly advocate neutralizing arguments and their snippets, we currently lack evidence that directly correlates (to a large extent) with satisfying users' information needs. Another unexplored aspect is to investigate whether the generation of snippets, especially through prompting, implicitly incorporates neutralization to some extent. These questions are subject to future research in the given context.

It is crucial to note that the success of generating and neutralizing snippets is closely linked to the quality of the original arguments. In cases where the original arguments are poorly constructed or unclear, the resulting objective snippets may not effectively represent their gist. We also recognize that neutralization is not appropriate in certain contexts where preserving the original language of the source text is critical (e.g., student essays, legal documents, or medical fields). In such cases, the application of neutralization requires the user's consent to ensure transparency and accountability. Practical implementations of our approach could include user options that allow individuals to choose between the original and neutralized versions of a snippet or an argument. We further acknowledge that our assumption that the generated arguments are gists of the original arguments may not always hold true. In some cases, the generated arguments may not capture the essence of the original arguments, leading to a loss of information.

We would like to acknowledge that the task of creating and neutralizing snippets is to a certain extent subjective. The choice of the best snippet may vary depending on the annotator's background, experience, and personal preferences. For this reason, we believe that further research is needed to explore the influence of these factors on the quality of the generated snippets and, in particular, to involve the authors of the original arguments in the process of snippet generation.

In summary, our empirical research highlights the potential benefits of mitigating subjective bias, particularly in the broader context of engaging with the opinions and arguments of others. This does not only facilitate informed decision making, but it can also be valuable for educational purposes.

6 Conclusion

In this paper, we have investigated the hypothesis that "objective snippets" of arguments are better for argument search engine results than state-of-the-art extractive snippets, using methods that combine snippet generation and neutralization. Our study has conveyed that a BART-based supervised summarization model outperforms a zero-shot Alpaca model to snippet generation. For neutralization, we have found that using reinforcement learning to align a large language model with human preferences for suitable arguments works best. We have also observed that both tasks complement each other and that their combination leads to the most effective snippets, as shown by human evaluation. Our results provide important insights and innovative methods that can be used to improve search engines in order to produce more efficient search results for users.

Acknowledgment. This project has been partially funded by the German Research Foundation (DFG) within the project OASiS, project number 455913891, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999), and by the EU, as part of the project OpenWebSearch.eu, grant agreement number 101070014, as part of the Horizon Europe research and innovation program.

A Appendix

Table 5. Examples of arguments from the appropriateness corpus and their neutralized snippets generated by our approach.

Type	Example
Argument	I always hear people talking about how the death penalty needs to be enforced more often. I, personally, am undecided on my opinion on the death penalty, but in my mind, the death penalty is used primarily as a societal form of retaliation (much the way that putting someone's head on a pike was used in the past). [...] At the very least, it will disseminate information to public about what execution in the United States is actually like (this may, in turn, change the beliefs that someone holds on the death penalty)
Neutralized Snippet	It is important for the public to be aware of the death penalty in the U.S. in order to make informed decisions about the practice. Televising executions could help to bridge the gap in understanding between the public and the criminal justice system.
Argument	At Poland Spring, we know that freshness and consistency go hand in hand. Our rigorous quality control standards help us make sure that you will enjoy the same crisp, delicious water every time you take a sip. To maintain the purity of our water, we employ a comprehensive, multi-barrier filtration system that involves carefully controlled and continuously monitored disinfection processes in hygienically designed lines. [...] Every year, we screen for over 200 possible contaminants-far more than state and federal regulations require
Neutralized Snippet	Poland Spring utilizes a rigorous filtration system, which is tested regularly throughout the production process to ensure the safety of the water. This process ensures the water quality remains consistent and meets strict quality standards.
Argument	CarCraze—you are wrong, sorry to say What you have there, my dear, is a bona fide Court Order for you to appear in court to answer to a Motion to Collect a Debt. [...] If you "just let it ride", as you say, don't be surprised when you open up your paycheck and you find a good chunk of it GONE ! What you should do: Seek legal counsel (get a lawyer) and come up with a manageable payment plan that is both satisfactory to your creditor, and workable for you. DON'T HESITATE - you will regret it
Neutralized Snippet	If you choose to ignore the notice of motion from the collection firm, it is possible that you may incur additional fees and charges. It is best to seek legal counsel from a professional in order to create a manageable payment plan that suits your financial situation.
Argument	In my opinion, all it serves to do is remind men to fulfil the traditional gender roles. To "be a man" you have to provide for your family, be responsible and protect your family. It's the male equivalent of "be more ladylike". All it accomplishes is: * Make non-traditionally masculine men feel like shit * [...] And god forbid that men have some values that women are 'supposed' to have. Why not have a gender neutral phrase? "Be a better person"?
Neutralized Snippet	This phrase can be harmful as it can lead to feelings of exclusion for men who don't conform to traditional gender roles, as these expectations can be seen as exclusive to men. This can be damaging as it may make those men feel like they do not belong

Table 6. A sample of comments provided by annotators organized by quality dimensions that influence snippet generation. Comments are edited for presentation purposes. Also, the anonymized snippets referred to in the comments (e.g., A, B, C) do not always correspond to a specific model being evaluated as the order of the snippets was randomized.

Dimension	Annotators' Comments
Conciseness	"This snippet provides a concise and clear definition of feminism, emphasizing equality and respect for both men and women."
Focus	"Snippet A is ranked highest for its clear emphasis on the necessity of abortion..."
Offensive Language	"While this snippet discusses activism against woman abuse and negative elements related to women, it introduces terms that may be considered offensive (e.g., "SLuts").", "The use of language like "I am going to have to ask you go to timeout because that idea is downright childish" might be perceived as confrontational."
Informativeness	The first snippet condenses the argument very succinctly and covers most of the major points in the arguments above
Structure & Coherence	"Snippet B is the worst summary for the argument presented above because there is not direct link between the statement and the conclusion of the snippet - so it completely misses the point.", "Snippet C is by far the best snippet in this sequence. It has a clear structure and it delivers the message of the paragraph."
Grammaticality	"Snippet A and B both have grammatical errors (need/needed), which would discredit the link/argument/page from the get go."
Self-contained	"Snippet A is ranked 3rd because there is no logical link between the first part of the snippet and the second part of the snippet. No reader could understand what the argument is about from that summary alone."
Accuracy	"Snippet C comes in last because it is completely inaccurate, given that it claims these points of view are Trump's points of view. In fact, they are the views of the narrator/author."
Argument-friendly Vocabulary	"Argument A has slightly more argument-friendly vocabulary (e.g' juxtaposition' used in contrast to'antithesis').", "The only main difference between argument A and C is the choice of vocabulary to describe the couples that the writer is associated with being either committed or monogamous. I think that the use of the word'committed' to describe the couple in argument A makes the example used more relevant to the argument at hand."
Seriousness	"Although argument A and C are similar, argument A has a more sincere tone and slightly more proper grammar: e.g. Latinos not "Latinos" / Latin America not Latinamerica."
Profanity & Speculation	"B is more to the point, doesn't speculate on strategies and has no profanity like C/B (Shit/assholes)."
Clarity	"This argument is clear in its meaning, provides a concise comparison between the two cases, and avoids inappropriate language or tone." , "This argument presents the issue clearly, maintains a proportional and balanced perspective by addressing both sides..."

Table 7. A sample of comments provided by annotators organized by quality dimensions that influence preference of snippets. Comments are edited for presentation purposes. Also, the anonymized arguments referred to in the comments (e.g., A, B, C) do not always correspond to a specific model being evaluated as the order of the arguments was randomized.

Dimension	Annotators' Comments
Respectful	"It begins with a dismissive tone ("is totally crap")...", "This argument is the best as it presents its points in a clear and respectful manner.", "This argument uses sarcasm ("(pause here for deeply bitter laugh)") and refers to a political figure in a dismissive manner ("The Shrub").", "This argument is the most appropriate as it maintains a professional tone, focuses on the key issues, and promotes a respectful and balanced discussion of the pro-choice vs. pro-life debate."
Critical Evaluation	"The mention of "corrupt the minds of my children" is emotionally charged, which may not provide room for critical evaluation.", "the ending part "expecting male users to do the looking for both themselves and the women" may come off as slightly dismissive, which makes it less open to others' arguments."
Formal Language	"This argument and Argument A are quite similar, but Argument C uses slightly more refined and formal language, making it more appropriate for a professional discussion. For example, it uses "Fourth Amendment" instead of "4th" and "naive" instead of "living under a rock"."
Grammaticality	"It has orthographic errors (e.g., missing spaces and inconsistent punctuation), making it harder to follow. Some of the phrasing is repetitive, and its presentation can hinder a clear understanding of its main points.", "...contains orthographic errors ("ur", "shld", "dugs"), and uses casual and unclear language. This decreases its credibility and appropriateness for a professional debate."
Conciseness	"I chose snippet A because it uses short sentences instead of one long one, and because it uses numbers which is more concrete than just saying "a high degree."", "Both very similar but snippet B is more concise..."
Evidence	"Snippet B is very subjective and doesn't present any evidence for the argument.", "...uses more numbers which encourages me to read.", "Snippet B is less pushy and provides more examples to back up its argument."
Grammaticality	"Snippet B has grammar and spelling errors which discourages me from wanting to read more."
Critical Evaluation	"A places the onus of thought on the reader, allowing them the space to form their own opinions. B is instructional, seemingly saying everything that is needed for a reader to make their mind up without their own research."
Structure & Coherence	"Snippet A clearly outlines their argument, while snippet B hops back and forth from one point to another without a linear thought process."

References

1. Ajjour, Y., Wachsmuth, H., Kiesel, J., Potthast, M., Hagen, M., Stein, B.: Data acquisition for argument search: the args.me corpus. In: Benzmüller, C., Stuckenschmidt, H. (eds.) KI 2019. LNCS (LNAI), vol. 11793, pp. 48–59. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30179-8_4
2. Alshomary, M., Düsterhus, N., Wachsmuth, H.: Extractive snippet generation for arguments. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2020, Virtual Event, China, 25–30 July 2020, pp. 1969–1972. ACM (2020). https://doi.org/10.1145/3397271.3401186

3. Bando, L.L., Scholer, F., Turpin, A.: Constructing query-biased summaries: a comparison of human and system generated snippets. In: Proceedings of the Third Symposium on Information Interaction in Context, pp. 195–204 (2010)
4. Bar-Haim, R., Eden, L., Friedman, R., Kantor, Y., Lahav, D., Slonim, N.: From arguments to key points: towards automatic argument summarization. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, 5–10 July 2020, pp. 4029–4039. Association for Computational Linguistics (2020). https://www.aclweb.org/anthology/2020.acl-main.371/
5. Cao, Z., Wei, F., Li, W., Li, S.: Faithful to the original: fact aware neural abstractive summarization. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th Innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, 2–7 February 2018, pp. 4784–4791, AAAI Press (2018). https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16121
6. Chen, P., Wu, F., Wang, T., Ding, W.: A semantic QA-based approach for text summarization evaluation. In: McIlraith, S.A., Weinberger, K.Q. (eds.) Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, 2–7 February 2018, pp. 4800–4807, AAAI Press (2018). https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16115
7. Chen, W., Syed, S., Stein, B., Hagen, M., Potthast, M.: Abstractive snippet generation. In: Huang, Y., King, I., Liu, T., van Steen, M. (eds.) WWW 2020: The Web Conference 2020, Taipei, Taiwan, 20–24 April 2020, pp. 1309–1319. ACM / IW3C2 (2020). https://doi.org/10.1145/3366423.3380206. https://doi.org/10.1145/3366423.3380206
8. Chen, W.F., Al Khatib, K., Stein, B., Wachsmuth, H.: Controlled neural sentence-level reframing of news articles. In: Moens, M.F., Huang, X., Specia, L., Yih, S.W.T. (eds.) Findings of the Association for Computational Linguistics: EMNLP 2021, pp. 2683–2693, Association for Computational Linguistics, Punta Cana, Dominican Republic, November 2021. https://doi.org/10.18653/v1/2021.findings-emnlp.228. https://aclanthology.org/2021.findings-emnlp.228
9. Croft, W.B., Metzler, D., Strohman, T.: Search Engines: Information Retrieval in Practice, vol. 520. Addison-Wesley Reading (2010)
10. Dai, N., Liang, J., Qiu, X., Huang, X.: Style transformer: unpaired text style transfer without disentangled latent representation. arXiv:1905.05621 [cs] (2019)
11. Dang, H.T.: Overview of DUC 2005. In: Proceedings of the Document Understanding Conference, vol. 2005, pp. 1–12 (2005)
12. Gero, K., Kedzie, C., Reeve, J., Chilton, L.: Low level linguistic controls for style transfer and content preservation. In: Proceedings of the 12th International Conference on Natural Language Generation, pp. 208–218 (2019)
13. Goyal, T., Li, J.J., Durrett, G.: News summarization and evaluation in the era of GPT-3. CoRR **abs/2209.12356** (2022). https://doi.org/10.48550/arXiv.2209.12356
14. Groeneveld, D., Meyerzon, D., Mowatt, D.: Generating search result summaries, uS Patent 7,853,587 (2010)
15. Hu, E.J., et al.: LoRA: low-rank adaptation of large language models. In: The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, 25–29 April 2022, OpenReview.net (2022). https://openreview.net/forum?id=nZeVKeeFYf9

16. Hu, Z., Yang, Z., Liang, X., Salakhutdinov, R., Xing, E.P.: Toward controlled generation of text. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 1587–1596 (2017)
17. Jin, D., Jin, Z., Hu, Z., Vechtomova, O., Mihalcea, R.: Deep learning for text style transfer: a survey. Comput. Linguistics (1), 155–205 (2022). https://doi.org/10.1162/COLI_A_00426
18. Kabbara, J., Cheung, J.C.K.: Stylistic transfer in natural language generation systems using recurrent neural networks. In: Proceedings of the Workshop on Uphill Battles in Language Processing, pp. 43–47 (2016)
19. Kendall, M.G.: Rank Correlation Methods (1948)
20. Krippendorff, K.: Estimating the reliability, systematic error and random error of interval data. Educ. Psychol. Measur. **1**, 61–70 (1970)
21. Lewis, M., et al.: BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, 5–10 July 2020, pp. 7871–7880. Association for Computational Linguistics (2020). https://www.aclweb.org/anthology/2020.acl-main.703/
22. Lin, C.Y.: ROUGE: a package for automatic evaluation of summaries. In: Text Summarization Branches Out, pp. 74–81. Association for Computational Linguistics, Barcelona, Spain, July 2004. https://www.aclweb.org/anthology/W04-1013
23. Madanagopal, K., Caverlee, J.: Reinforced sequence training based subjective bias correction. In: Vlachos, A., Augenstein, I. (eds.) Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pp. 2585–2598. Association for Computational Linguistics, Dubrovnik, Croatia, May 2023. https://doi.org/10.18653/v1/2023.eacl-main.189. https://aclanthology.org/2023.eacl-main.189
24. Maynez, J., Narayan, S., Bohnet, B., McDonald, R.T.: On faithfulness and factuality in abstractive summarization. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J.R. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, 5–10 July 2020, pp. 1906–1919. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.acl-main.173
25. Nallapati, R., Zhou, B., dos Santos, C.N., Gülçehre, Ç., Xiang, B.: Abstractive text summarization using sequence-to-sequence RNNs and beyond. In: Goldberg, Y., Riezler, S. (eds.) Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, 11–12 August 2016, pp. 280–290. ACL (2016). https://doi.org/10.18653/v1/k16-1028
26. Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank citation ranking: bringing order to the web. Technical report, Stanford InfoLab (1999)
27. Pryzant, R., Martinez, R.D., Dass, N., Kurohashi, S., Jurafsky, D., Yang, D.: Automatically neutralizing subjective bias in text. In: Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence, pp. 480–489 (2020)
28. Stab, C., et al.: ArgumenText: searching for arguments in heterogeneous sources. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 21–25. Association for Computational Linguistics, New Orleans, Louisiana, June 2018. https://doi.org/10.18653/v1/N18-5005. https://www.aclweb.org/anthology/N18-5005
29. Stiennon, N., et al.: Learning to summarize from human feedback. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS 2020, Curran Associates Inc., Red Hook, NY, USA (2020). ISBN 9781713829546

30. Syed, S., Khatib, K.A., Alshomary, M., Wachsmuth, H., Potthast, M.: Generating informative conclusions for argumentative texts. In: Zong, C., Xia, F., Li, W., Navigli, R. (eds.) Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, 1–6 August 2021, Findings of ACL, vol. ACL/IJCNLP 2021, pp. 3482–3493. Association for Computational Linguistics (2021). https://doi.org/10.18653/v1/2021.findings-acl.306
31. Taori, R., et al.: Stanford Alpaca: an instruction-following LLaMA model (2023). https://github.com/tatsu-lab/stanford_alpaca
32. Tombros, A., Sanderson, M.: Advantages of Query Biased Summaries in Information Retrieval. In: Proceedings of SIGIR 1998, pp. 2–10 (1998)
33. Wachsmuth, H., et al.: Building an argument search engine for the web. In: Proceedings of the 4th Workshop on Argument Mining, pp. 49–59. Association for Computational Linguistics, Copenhagen, Denmark, September 2017. https://doi.org/10.18653/v1/W17-5106. https://www.aclweb.org/anthology/W17-5106
34. Walker, M.: Information and deliberation in discourse. In: Intentionality and Structure in Discourse Relations (1993). https://aclanthology.org/W93-0238
35. White, R., Ruthven, I., Jose, J.M.: Web document summarization: a task-oriented evaluation. In: 12th International Workshop on Database and Expert Systems Applications, pp. 951–955. IEEE (2001)
36. Zhang, J., Zhao, Y., Saleh, M., Liu, P.J.: PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In: Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13–18 July 2020, Virtual Event, Proceedings of Machine Learning Research, vol. 119, pp. 11328–11339. PMLR (2020). http://proceedings.mlr.press/v119/zhang20ae.html
37. Ziegenbein, T., Skitalinskaya, G., Makou, A.B., Wachsmuth, H.: LLM-based rewriting of inappropriate argumentation using reinforcement learning (2024)
38. Ziegenbein, T., Syed, S., Lange, F., Potthast, M., Wachsmuth, H.: Modeling appropriate language in argumentation. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL), pp. 4344–4363. Association for Computational Linguistics, July 2023. https://doi.org/10.18653/v1/2023.acl-long.238. https://aclanthology.org/2023.acl-long.238.pdf
39. Ziegler, D.M., et al.: Fine-tuning language models from human preferences. CoRR (2019). http://arxiv.org/abs/1909.08593

ArgServices: A Microservice-Based Architecture for Argumentation Machines

Mirko Lenz[1(✉)], Lorik Dumani[1], Ralf Schenkel[1], and Ralph Bergmann[1,2]

[1] Trier University, Universitätsring 15, 54296 Trier, Germany
info@mirko-lenz.de, {lenz,dumani,schenkel,bergmann}@uni-trier.de
[2] Branch Trier University, German Research Center for Artificial Intelligence (DFKI), Behringstr. 21, 54296 Trier, Germany
ralph.bergmann@dfki.de

Abstract. Argumentation is ubiquitous, and the development of argumentation machines could greatly assist humans in managing and navigating argumentation. However, the development of such systems is hindered by the lack of common standards and suitable tools, leading to ad-hoc solutions with little reuse value. Towards a more unified approach, we present an extensible microservice-based architecture for argumentation machines. Being built on the established gRPC framework, it provides strongly typed interfaces for the following services: (i) Argument Mining, (ii) Case-Based Reasoning on Arguments, (iii) Argument Retrieval and Ranking, and (iv) Quality Assessment of Arguments. Our system is designed to be extensible, allowing for easy integration of new tasks. We demonstrate the feasibility of our architecture via a proof-of-concept implementation and provide additional supplementary resources, such as a REST API gateway. Our contributions are publicly available on GitHub under the permissive MIT license.

Keywords: Argumentation · Argument Graphs · Microservices · gRPC · REST · Natural Language Processing · Open Source

1 Introduction

Living in an ever-changing world, we are constantly confronted with new information and, based on it, have to make decisions. With the advent of the Internet, computers have become an integral part of this process. While traditional Web search engines mostly rely on textual similarity, and thus require users to manually extract and analyze relevant information, domain-specific systems could incorporate other sources of knowledge to provide assistance. However, even within a single field—such as argumentation—many competing solutions exist without a common standard or interface. As a result, the development of ad-hoc solutions is often necessary, which are not easily reusable in other contexts. Argumentation machines [30] for example may offer a variety of services

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 352–369, 2024.
https://doi.org/10.1007/978-3-031-63536-6_21

like retrieval and validation, but contributions in the domain of Computational Argumentation (CA) focus mainly on one aspect of such a system.

In this paper, we present a microservice-based architecture for argumentation machines designed to be extensible and reusable. The target audience is researchers and developers who aim to build, extend, or only use specific functions of argumentation machines, without reinventing the wheel. Our ultimate vision is to allow other researchers to rewrite one service using their own algorithms and integrate it into the existing architecture—enabling them to evaluate their approach in a larger framework. Our contributions are (i) service descriptions for common tasks in CA based on the established gRPC framework, (ii) a proof-of-concept implementation of the architecture showcasing its feasibility, and (iii) a collection of supplementary resources like a REST API gateway and ready-to use client/server libraries.

The remainder of this paper is structured as follows: In Sect. 2, we introduce the foundations necessary to understand our architecture, followed by a discussion of related work in Sect. 3. Section 4 presents the service definitions that are implemented in a proof-of-concept described in Sect. 5 and supported by supplementary resources introduced in Sect. 6. Finally, Sect. 7 discusses current limitations and Sect. 8 concludes our paper.

2 Foundations

Our proposed architecture for an argumentation machine is fundamentally based on argument graphs, so we briefly introduce them in this section. Since dealing with texts is essential in this domain, we also present some common Natural Language Processing (NLP) [5] concepts here. Lastly, we introduce some aspects of microservice-oriented backends like Representational State Transfer (REST) and gRPC.

2.1 Theoretical Argumentation and Argument Graphs

An argument is typically composed of a single *claim* that is supported or attacked by one or multiple *premises* [28]—these smallest units of an argument can be subsumed under the term Argumentative Discourse Units (ADUs) [28]. A claim itself may also support another claim and thus additionally act as a premise, making it possible to represent entire conversations. Moreover, an argument typically has one primary/central conclusion, the so-called *major claim*. This inductive structure already forms a directed graph—we call it *argument graph*.

According to Argument Interchange Format (AIF) [13], an argument graph is a tuple $G = (V, E)$ where V is a set of nodes and $E \subseteq V \times V$ is a set of directed edges. The nodes are divided into atom nodes $A \subset V$ representing the ADUs and scheme nodes $S \subset V$ representing the relationships between them: $V = A \cup S$. Edges cannot be drawn between two atom nodes, so we define $E \subseteq V \times V \setminus A \times A$. Any sequential ordering of ADUs originating from the source text is lost in the AIF graph representation. To mitigate this, the annotation software Online

Visualization of Arguments (OVA) [8] for example uses additional properties to store the position of each atom node in the text and consequently the order of the ADUs. The creation of structured argument representations—including but not limited to graphs—is also known as Argument Mining (AM) [20].

To use more detailed semantics when representing the scheme nodes, argumentation schemes introduced by Walton et al. [41] may be used. Each scheme—for instance, *Expert Opinion*—explicitly describes the role of a claim and its fixed set of premises. In this example, the claim would be the conclusion of one premise representing an expert's opinion and the other premise representing the expert's expertise. To check the applicability of such a scheme to a relationship, the authors defined critical questions.

2.2 Argument Processing

Argument graphs contain two types of information—structure and semantics. The former refers to the graph-based representation (i.e., the nodes and edges), whereas the latter refers to the text of the nodes. Our system deeply integrated both aspects, so the following section will introduce the necessary concepts.

For the *structural* aspect, we use the wide variety of research on graph-based representations. A relevant field is Process-Oriented Case-Based Reasoning (POCBR), a variant of Case-Based Reasoning (CBR) [2,33] that focuses on graph-based workflows (e.g., business processes). The idea of CBR is to solve new problems by reusing solutions to problems similar to those that have been solved in the past by performing four steps: (i) *Retrieve* a set of similar cases from the so-called *case base*, (ii) *reuse* the found cases by *adapting* them to the new query, (iii) *revise* the adapted cases by checking their validity, and (iv) *retain* the new solution for future use. One central difference between case-based retrieval and Information Retrieval (IR) is the inclusion of structural information—in our case the argument graph.

When assessing the similarity between the atom nodes of two argument graphs in the CBR framework, we need to take into account the *semantic* aspect. Over the past few years, the use of language models—for instance, to compute the semantic similarity between texts via embeddings—has become a common practice in NLP. The basic idea is that words or sentences with similar meanings should be close to each other in a high-dimensional vector space. Using standard measures like the cosine distance, we can assess the similarity between two texts by comparing their embeddings/vectors. For the nodes of the scheme, embeddings could also be computed to assess their similarity, but taxonomy-based measures may be a better fit [40].

2.3 System Architectures

When designing a system, there are two common approaches: *monolithic* and *microservice-based* architectures. A monolith is a single codebase that contains all functionality of the system and is deployed as a single unit. On the contrary, a microservice-based architecture is composed of multiple coherent services that are deployed independently of each other. [4] In this paper, we committed to the latter for two main reasons: (i) The specific implementation of a single module is entirely separated from the rest, meaning it is possible to combine different programming languages (e.g., Java and Python). (ii) We aim at providing an argumentation machine that allows other researchers to quickly swap out a single module with their own implementation and evaluate it in a larger context. Although both can in principle be achieved with a monolithic architecture, the microservice-one was the more natural choice for us, since general-purpose monoliths are not yet widely used (see Sect. 3). In addition, a microservice architecture makes it easier to scale horizontally, which means that it translates well into production environments. In the following, we briefly introduce this architecture in more detail.

According to Jamshidi et al. [18], a service/module of a microservice-based system offers "access to its internal logic and data through a well-defined network interface"—the so-called Application Programming Interface (API). As of 2024, the most common style of these systems is REST, which uses a fixed set of URL-based endpoints and Hypertext Transfer Protocol (HTTP) operations to access the functionality of a service. There are also other options like Simple Object Access Protocol (SOAP), GraphQL, and gRPC, each having its own set of advantages and drawbacks. For our architecture, we ultimately settled on gRPC—a Remote Procedure Call framework developed by Google on top of the modern HTTP/2 protocol specifically for microservice backends.[1] Compared to the established REST, it has the following differences:

Stronger Typing. gRPC uses Protocol Buffers (Protobuf) for data serialization, which allows for a more strict definition of the data types used in the API. This strong contract between client and server removes some potential sources of bugs (e.g., sending strings instead of integers).

Code Generation. The use of Protobuf to define services provides a code generation tool that creates client and server stubs for most major programming languages. This means that compared to REST, the user does not have to deal with the low-level details of the HTTP protocol.

Binary Data Transfer. The messages sent between the client and the server are encoded in a binary format, which is much more compact than the textual JavaScript Object Notation (JSON) format used by REST. Note that this does not have a negative impact on readability, as it only applies to the transfer itself—that is, any Protobuf message can be serialized to a JSON object as well.

These advantages come at the cost of a steeper learning curve—for instance, developers need to learn a new domain-specific language and have to re-run the code generation tool after changes to the service definitions. With its reliance on HTTP/2, gRPC cannot be natively used in browsers and requires proxies to work around this limitation (see Sect. 6 for our solution). Yet, due to the mentioned advantages, gRPC and Protobuf are already heavily used in Machine

[1] https://grpc.io.

Learning (ML)—most prominently, they serve as the backbone for the official Tensorflow API.[2]

3 Related Work

To the best of our knowledge, there is almost no work on the architecture of argument machines as we present them. Works up to the mid-2010s often only presented computational models of argument which are more concerned with argumentation theory—that is, the construction of arguments or a whole argumentation. In the following, we consequently not only collected works describing entire argumentation machines, but also works concerned with only one aspect of it (e.g., argument retrieval).

The development of argumentation machines began in the mid-1990s and included work on argumentative dialog planning [29], applications of argument schemas in Artificial Intelligence (AI) [31], and argumentation engines capable of handling a large number of topics [30]. The AIF (see Sect. 2.1) was later extended to handle dialogical argumentation [32]. In the following years, the argument annotation tools ArgueBlogging [9] and OVA+ [19] were developed. They specialized in constructing discussions about blogs and annotating plain texts with argument graphs, respectively.

Slonim et al. [36] presented Project Debater, an autonomous debating system that is capable of discussing with people a wide variety of topics and taking certain positions. Even before the era of Large Language Models (LLMs), their system was capable of conducting a discussion—that is, understanding users' viewpoints and generating suitable arguments. In their work, they described the architecture of the system and conducted an evaluation that included several debate topics. An alternative architecture for an argumentation machine has been proposed by Bergmann et al. [6] as part of the ReCAP project. In our work, we build on their proposal and present an improved version that has been developed according to best practices in an effort to keep up with the rapidly changing field of CA.

Apart from this, we are only aware of work on stand-alone systems, which we present in the following. Wachsmuth et al. [39] proposed the first argument search engine (Args) known to us, which introduced a system that reads any free text user queries to search for arguments, and then presents relevant arguments from a pool of almost 300k previously mined and indexed arguments from five debate portals (i.e., Web content) in a ranking based on BM25F. Other projects used their system as a starting point for their own research—for instance, by reimplementing its most important properties or using their dataset for tasks like ranking [3,11]. However, despite all merits, their system does not cover the entire argumentation machine as we envision it.

Among others, Bondarenko et al. [11] have been setting up the CLEF lab Touché every year since 2020, where they used the Args dataset until 2022

[2] https://github.com/tensorflow/serving.

and ClueWeb22 since 2023. The systems submitted by the participants can be uploaded to TIRA [16] to reproduce the results. Like Args, ArgumenText [37] is an argument search engine. First, it finds relevant documents in a large set of heterogeneous arbitrary Web sources using ElasticSearch, then it identifies relevant premises in them using Keras, assigns them stances by applying BiLSTM, and ranks the premises by their classifier's confidence score. Its evaluation showed a high recall of 89% and a rather moderate precision of 47%.

Beyond the retrieval of arguments, Eden et al. [15] presented insights on the creation of a Key Point Analysis (KPA) system and highlighted some of the main challenges. KPA is concerned with extracting the main points from a collection of opinions, a service that may in the future also be incorporated into our proposed architecture. Romberg [34] tackles the problem that argumentation is often subjective and annotations are summarized with average or majority vote, resulting in minorities being ignored when learning. Therefore, she introduced PerspectifyMe, a method that combines subjective points of view by complementing an aggregated label with a subjectivity score. Heinisch et al. [17] addressed the subjectivity issue in annotation processes and found that classifiers incorporating relations between different annotators are beneficial even for predicting single-annotator labels. Building models that are aware of potentially subjective annotations is a crucial aspect in CA, so we plan to include this aspect in future iterations of our architecture.

4 Microservices for Argumentation

As mentioned in Sect. 1 and seen in Sect. 3, argumentation machines can vary greatly w.r.t. their functionality. Consequently, the main goal of our service definitions is to be easily extensible for tasks not envisioned by us. As a starting point, we identified the following tasks as common in CA: (i) argument mining, (ii) case-based reasoning on arguments, (iii) retrieval and ranking of arguments, and (iv) quality assessment of arguments. Please note the difference between (ii) and (iii): while the former integrates the structural information of entire graphs (see Sect. 2.2), the latter considers ADUs or claim-premise pairs. An overview of the different modules is given in Fig. 1. There are two special services in our proposed architecture that were not part of the aforementioned list: (i) The argumentation base in the middle and (ii) the NLP service to the bottom right. All services either consume argument graphs or produce them, so we created a dedicated module to serve them. Some of the services also require NLP functionality, leading to the creation of a separate service for this task. All services are designed to work independently of each other (with the exception of the NLP service), but may be combined to form a complete argumentation machine. Further aspects of their orchestration are discussed in Sect. 5.

All services are defined using Protobuf and gRPC with their definitions publicly available on GitHub under the permissive MIT license.[3] These service definitions are also available from the Buf Schema Registry, which makes it possible

[3] https://github.com/recap-utr/arg-services.

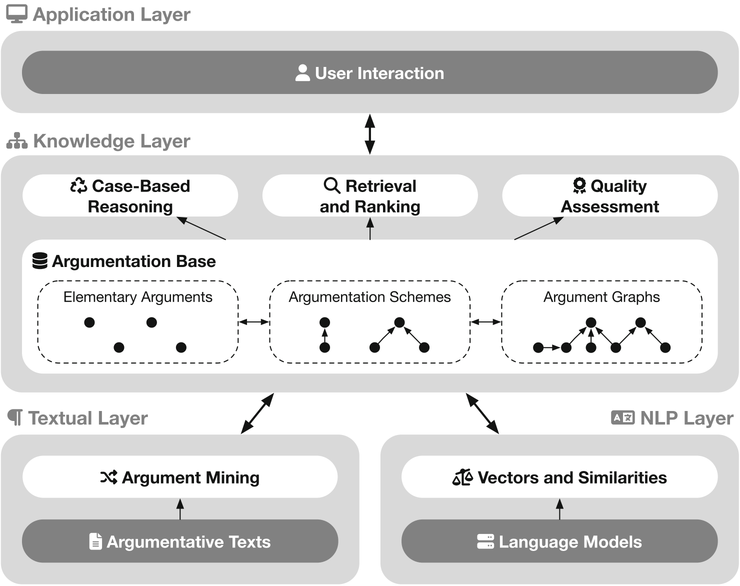


Fig. 1. Overview of our proposed architecture for microservice-based argumentation machines. Light gray boxes represent the different layers, dark gray ones external resources, and white ones the services exposed via our API.

to add them as dependencies in other gRPC-based project and provides users with a nicely formatted and up-to-date documentation.[4] We acknowledge that there may be varying requirements or the need for additional data depending on the context. Consequently, each function described here allows arbitrary JSON-encoded data to be encoded as an optional parameter called *extras*.

In the following section, we introduce each service in detail and provide a list of all included functions. We also highlight some of the most important options/parameters that can be used to customize the behavior of the services. Due to their tight integration with the other services, we start with two core modules: the argumentation base (middle) and NLP service (bottom right). Subsequently, the remaining services mentioned above are introduced.

4.1 Argumentation Base

The core of our argumentation base is our own argument serialization format Argument Buffers (ARGUEBUF) [21] first introduced at COMMA 2022. Given that it has been designed as a first-class citizen of Protobuf, the integration into our architecture is straightforward. Its formal semantics are based on the established AIF standard (see Sect. 2.1), but the storage format is designed to be more uniform and extensible at the same time. With regards to the *graph*

[4] https://buf.build/recap/arg-services.

structure, the following five main differences exist compared to argument graphs serialized to AIF: (i) The graph itself and each of its nodes/edges allow arbitrary key-value pairs to be stored. (ii) The sets containing the nodes and edges are represented as dictionaries with their respective IDs as keys to enforce uniqueness. (iii) Original textual resources and participants of a conversation can be stored together with the graph. (iv) The major claim is explicitly marked as such. (v) Information about analysts and the creation/modification date can be stored. *Atom nodes* can not only store the text of the ADU, but also the position in the original text and the participant who made the statement. The link to the original text also allows for reconstruction of the sequential ordering of the argument as found in the source text. *Scheme nodes* do no longer use a free-text field to store the scheme name, but instead refer to a scheme from a predefined list through an enumeration. This decision makes parsing and serialization easier and more reliable, with the drawback that new schemes first need to be added to the Protobuf definition. All the mentioned changes are additions to the format, so an existing AIF graph can easily be converted to Arguebuf. An area where our format is currently lacking is the representation of dialogical argumentation, which is planned to be added in the future. This service offers a single function:

Casebase. Given a list of filter criteria (expressed as regular expressions), return a list of argument graphs. The type of filters available depends on the implementation—thus the use of generic regexes—but may include factors like the corpus name, the serialization format, or the inclusion of schemes.

We have chosen to stick to the CBR terminology here to be consistent with some of our other services. Currently, only regex-based filtering is supported, but more advanced filtering options could be added in the future if the need arises.

4.2 Natural Language Processing

As outlined earlier, the use of language models—for instance, to compute the semantic similarity between texts via embeddings—has become a common practice in NLP. At the same time, these models are getting larger and larger, making it harder to use them on a regular computer. When combined with a microservice architecture, another challenge is that each service would need to load the model into memory, which is a waste of resources. We therefore chose to add a dedicated service for these needs with a central NlpConfig message that can be passed between individual services. It encodes (i) the language of the processing pipeline, (ii) the choice of the language model (multiple are also possible), (iii) the similarity measure to use, and (iv) the pooling function for plain word embeddings. Currently, this service focuses on determining semantic similarity between texts through embeddings. More general NLP tasks like named entity recognition or dependency parsing are not yet supported but could be added in the future if the need arises. To mitigate this restriction, we added a function to process texts with the Python library spaCy [25] and return the result as a binary representation. Please note that the goal of this service is to save resources by loading common base models once instead of requiring each service to load them individually. That also means that custom models (e.g., for classification) are not part of this service and need to be handled by the respective service implementations. It is also beyond the scope of this service to add generative models, as there already exist well-established interfaces like the OpenAI API for this purpose. The corresponding service offers the following functions:

Vectors. Given a list of texts, it returns their embeddings of n dimensions.

Similarity. Given a list of text pairs, it returns their similarity score between 0 and 1.

Spacy Document. Give a list of texts, return the corresponding spaCy documents. This is an optional service that is usable only with Python servers and clients.

4.3 Argument Mining

Having introduced the two cornerstones of our architecture, we now present the service responsible for building our argumentation base through AM. The set of functions is derived from an end-to-end pipline [23] for transforming plain texts into argument graphs consisting of multiple successive steps with an additional function for transforming a text to a graph without any intermediate steps.

Segment Text. Given a natural language text, return the list of Elementary Discourse Units (EDUs).

Classify ADUs. Given a list of EDUs, return the list of ADUs.

Predict Major Claim. Given a list of ADUs, return a ranking of major claim candidates.

Predict Polarity/Entailment. Given a list of ADUs, compute the cross product to generate claim-premise pairs and predict the polarities (i.e., support, attack, or neutral) between them.

Construct Graph. Given all ADUs, the major claim, and the predicted polarities, construct the resulting graph using some heuristic.

End-to-End Pipeline. Given a natural language text, return an argument graph.

With the advent of generative language models, we plan to add a function to the service that allows the generation of textual arguments and/or argument graphs from a given prompt. Although even the present functions can already be implemented using LLMs, the current set of features is more focused on the extraction of arguments from existing texts. As such, the envisioned generation function would enable the synthesis of new arguments.

4.4 Case-Based Reasoning on Arguments

With the methods in place to build the argumentation base, we have now reached the knowledge layer and can take advantage of them in our services. The retrieval functionality is motivated by our paper published at FLAIRS 2019 [7] that proposes a combination of semantic and structural similarity measures for CBR with argument graphs. The underlying paper for the adaptation service has been published at ICCBR 2023 [22] and proposes a hybrid approach combining WordNet with LLMs to adapt retrieved arguments. For all methods offered by the service, the cases and the user-provided are represented as argument graphs—enabling the user to specify the structure and the content of the desired argument. It offers the following functions:

Retrieve. Given a collection of argument graphs (i.e., the casebase) and a user-defined query (that is also a graph), perform a search for the most similar argument in the casebase.

Adapt. Given one retrieved graph and the user-defined query, perform a keyword-based adaptation with the goal of making the retrieved one more similar to the query. The function allows passing one or multiple rules to influence the process. One can decide to restrict the adaptation process to pure generalization, pure specialization, or a combination of both.

4.5 Argument Retrieval and Ranking

Argument Retrieval contains a wide range of IR tasks, including ranking and clustering—the former being at the heart of every IR system. At SIGIR 2021 [27] we presented an argument search system that ranks premises to queries according to the principle of TF-IDF (i.e., the more frequent premises of claims that are (more) similar to the query occur, the higher the score), as well as by the three (main) quality dimensions of cogency, reasonableness, and effectiveness [38]. At CIKM 2021 [14] we presented a work on fine granular clustering of arguments, as clustering is an essential part of our ranking approaches. The service offers the following functions:

Statistical Ranking. Given a query and a list of ADUs, return a ranking of the given arguments based on frequency and specificity.

Quality-Based Ranking. Given a query and a list of ADUs, return a ranking of the given arguments based on scores derived from a set of quality dimensions.

Fine-Granular Clustering. Given a query and a list of ADUs, predict a set of scores used to assign them to fine-granular clusters.

4.6 Quality Assessment of Arguments

As implied in the previous section, we used argument quality in our work, which is why we also offer a dedicated service for this task. A work presented at CIKM 2023 [12] introduced a User Interface (UI) that takes two premises for a claim and not only decides for these two, which is more convincing for all 15 argument quality dimensions [38], but also provides an additional explanation together with the individual scores justifying why the particular decisions were made. Another work published at the ArgMining workshop at COLING 2022 [10] presented an end-to-end tool that reads any plain text and returns the so-called qualia structures (which express the meaning of lexical items from four viewpoints). With validation being a central part of argument quality—arguments containing disinformation have lower quality—we also provide a further service that is based on a work presented at the TMG workshop at ICCBR 2023 [26] able to predict the suitability of experts when cited for emphasizing statements. The quality service has the following functions:

Quality Explanation. Given a claim and two premises, determine and explain which one is more convincing for all available quality dimensions as well as globally across all dimensions.

Qualia Annotations. Given a text and a list of qualia patterns, compute the constituency tree and return the qualia role for each pattern.

Expert Suitability. Given a premise and (optionally) the Google Scholar ID of a researcher, predict whether they are an expert on the given topic.

5 Proof-of-Concept

With all the necessary tools in place, we created a proof-of-concept implementation of our architecture that includes almost all the services and functions described in Sect. 4. One of the central goals of our work has been to create a machine that allows other researchers to reimplement a single module and gain the ability to perform experiments in a larger system. Consequently, some of the services are written in Python, while others use Java. The code is based on existing implementations originally written for the corresponding paper or was newly created for this work. Some services are even implemented through a LLM-based prompting strategy to showcase the flexibility of our architecture in keeping up with the latest trends in NLP. The code and additional instructions are available on GitHub under the permissive MIT license.[5] In the following section, we present individual service implementations and discuss their orchestration. To wrap up, we also introduce an evaluation framework for the CBR services that demonstrates the client side of our architecture.

Argumentation Base. The argumentation base is provided by our ARGUEBUF Python library (see Sect. 6 for more details). It can serve argument graphs from a local directory or a remote server and allows to filter them with regular expressions. Our implementation expects that filter criteria are stored in directory names using the pattern `<property1>=<value1>,<property2>=<value2>,...` and therefore allows the use of arbitrary properties for filtering.

To get started more easily, we provide a public collection of argument graphs called ARGUEBASE[6] that adheres to the naming convention mentioned of the directory. It contains a diverse set of publicly argument graph corpora in various formats like AIF or ARGUEBUF and includes links to the original sources and licenses.

[5] https://github.com/recap-utr/arg-services-poc.

[6] https://github.com/recap-utr/arguebase.

Natural Language Processing. Our NLP service implementation is written in Python and built on the popular spaCy library—including a specialized client to simplify the consumption of the service. Besides the embedding models offered by spaCy, our services provide an integration with SentenceTransformers[7] that allows to use a wide range of pre-trained models for computing contextualized embeddings and includes support for CUDA acceleration. For regular word embeddings, multiple pooling methods are supported in addition to the default pooling of the mean, including the generalized power mean [35]. Instead of applying cosine similarity to pooled vectors, max-pooling can be used to determine the similarity between two texts [42]. Finally, multiple models can be selected at the same time with their individual embeddings concatenated to a larger vector.

Argument Mining. Based on a prompting strategy created for another project (currently in development), we implemented an LLM-based AM service in Python. The service allows to run the stages individually or as an end-to-end pipeline and makes use of the *function calling* feature of OpenAI's ChatGPT to enforce a JSON schema for the predictions. It demonstrates that even in light of generative models getting better at many tasks, our architecture can act as a *translation layer* between new and existing systems.

Case-Based Reasoning on Arguments. The retrieval functionality is implemented in the Python application ArgueQuery and uses both semantic and structural similarity measures for CBR with argument graphs. The semantic part is handled by comparing embeddings, while the structural part involves an A* search algorithm to find the best mapping between the user query and the graphs in the case base.

Adaptation is possible through ArgueGen and uses a combination of WordNet [1,24] and LLMs to adapt retrieved arguments. For each argument to be adapted, the Python-based service first identifies the central keywords, prompts a generative language model for suitable replacements, verifies the response using the WordNet database, and applies all valid ones to the argument graph. In addition to this hybrid approach, it is also possible to perform the adaptation solely based on LLMs or WordNet.

Both of these services make use of the NLP service of our architecture to compute the embeddings and extract other linguistic features from the texts. They show the power of the NlpConfig message: Each service receives an NLP configuration object containing parameters like the model to use, and then uses it themselves to perform requests to the NLP service. In this way, it is possible to share a single object between all services but also use different configurations for certain services if needed.

Argument Retrieval and Ranking. For the two ranking functions, a Java-based application built on Apache Lucene[8] is available. Given a user-provided

[7] https://www.sbert.net/.

[8] https://lucene.apache.org/.

textual query, this argument search engine returns a ranked list of representatives from premise clusters. It first searches an inverted index for claims that are similar to the query, identifies all linked premises (pre-clustered according to their semantics), and then ranks these clusters at runtime using frequencies or argument quality. For the fine-granular clustering function, we developed a prompt-based strategy leveraging OpenAI's ChatGPT to provide responses (similar to the argument mining service).

Quality Assessment of Arguments. The same LLM-based approach was used to provide a prototype of our quality explanation function. To determine the qualia annotations, a text and a list of patterns consisting of sequences of POS tags are expected. The Java-based system then creates the constituency trees of the text and searches for the patterns. If these are found, the qualia role and the qualia query that match a pattern are output for each match. The expert suitability function is the only functionality of our proposed architecture that is not part of this proof-of-concept.

Orchestration of Services. The ultimate goal of our architecture is to provide an integrated argumentation machine that exposes a set of services to the user. To simplify the deployment and orchestration of these services, we provide Docker-based containers for many of the services described in this section. These containers can be managed jointly using Docker Compose, for which we provide a configuration template as part of our proof-of-concept implementation. For our two central services—that is, the argumentation base and NLP service—we also provide pre-built images that can be pulled directly from the GitHub Container Registry. For all argument mining and CBR services, we created ready-to-use Docker files that can be built locally and integrated into the Docker Compose configuration. The services for ranking and quality assessment of arguments (see Sects. 4.5 and 4.6) are mostly written in Java and require custom binary files, so their deployment is more involved. We plan to provide Docker images for the in the future as well.

Evaluation Client. To evaluate our CBR services, we created a client called ArgueLauncher[9] that can be used to compare the results of the argumentation machine to a gold standard. It is written in Python, can be used to evaluate the retrieval and adaptation services, and contains an abstract interface for evaluations that can be easily extended to other services in the future. With this application using only the client libraries of the services, it can also be used by other developers as a starting point for integrating our architecture into their own systems.

[9] https://github.com/recap-utr/arguelauncher.

6 Supplementary Resources

Besides the service descriptions and their accompanying Protobuf definitions, we also provide a few additional tools and resources to simplify the development of argumentation machines. When designing these, we strived to follow the best practices of software engineering. For instance, all libraries strictly adhere to the semantic versioning scheme and provide a changelog for each release. We also make extensive use of the package manager *Nix*[10] to manage our dependencies and provide reproducible environments. New releases are published through a Continuous Integration (CI) pipeline that leverages our Nix setup in GitHub actions. In the next section, we highlight what we believe are the most useful resources for other researchers and developers in the domain of CA.

Ready-to-Use Client and Server Libraries. As stated in Sect. 2.3, Protobuf offers easy code generation of native libraries for most programming languages, but this additional step may already be a hurdle for some developers. To lower the barrier of entry, we provide ready-to-use client and server libraries for Python, TypeScript/JavaScript, and Java.[11] They can be installed via their native package manager—that is, `pip`, `npm`, and `maven`.

Creating a new argument graph format for our microservices allows first-class support without the need for numerous format conversions. However, it also means that existing and commonly used formats like AIF cannot be used out-of-the-box. To remedy this, we provide *supercharged* libraries for Python and JavaScript/TypeScript that make it easy to import graphs in AIF, Argdown, Kialo, OVA3, SADFace, and xAIF and export them to AIF and xAIF.[12] They also contain optimized graph representations that abstract some Protobuf-specific details away and make it easier to work with argument graphs in these languages. The Python version is additionally integrated with NetworkX and can render images of graphs using D2 and Graphviz.

REST API Gateway. Even though gRPC provides major advantages over REST and we try to reduce the burden of using it as much as possible, it is still not as widely used as REST. It may also be the case that a developer wants to integrate our services into an existing system that already uses REST APIs. To combine the best of both worlds, we created a proxy[13] that allows REST clients to access any gRPC service. It is based on the popular Envoy proxy[14] and is provided as a Docker image and a binary file for all three major operating systems: Windows, Linux, and macOS. Additionally, it also supports the conversion between gRPC-Web requests and regular gRPC requests (which is needed for browser-based clients).

[10] https://nixos.org.

[11] https://github.com/recap-utr/arg-services.

[12] https://github.com/recap-utr/arguebuf.

[13] https://github.com/mirkolenz/grpc-proxy.

[14] https://www.envoyproxy.io/.

Argument Mapping Interface. First introduced at COMMA 2022 [21], our tool ArgueMapper [21] is a web-based interface for creating argument graphs.[15] Compared to established solutions such as OVA, it uses a modern development stack (TypeScript and React) and is thus easier to extend and maintain. ArgueMapper natively support our Protobuf-based serialization format Arguebuf and can be used to build the argumentation base for our microservices.

7 Limitations

While we have tried to create an extensible architecture that can be used for a wide range of tasks in CA, there are still some limitations to our approach. First, the reliance on gRPC and Protobuf may be a hurdle for some developers: Although they provide a strong contract between the client and the server, they are not as widely used as REST—especially in the context of research projects. We try in part to mitigate this through our REST API gateway, but it is still an additional layer that needs to be managed.

The representation of arguments through graphs is backed deeply into our architecture and may not be suitable for all argumentative domains. For example, some texts may contain arguments that are only loosely connected or where the relations between them are implicit—like in news editorials. In such cases, the strict graph structure may not be the best choice for representing arguments. In addition, our Arguebuf format does not yet support dialogical argumentation, limiting its applicability in this domain.

Lastly, the evaluation of our architecture is still ongoing, and we have not yet tested it in a real-world scenario. We assume that there exist some features that are missing and/or incomplete when coming to new domains, but we are open for feedback and contributions from the community to improve our architecture—which is possible due to the forward- and backward-compatibility of Protobuf. An example of such missing features is our NLP service that is currently focused only on the extraction of linguistic features and the computation of semantic similarity. In the future, it may be desirable to extend its scope and integrate functions to serve custom models to other services.

8 Conclusion and Future Work

In this paper, we presented an extensible microservice-based architecture for argumentation machines based on gRPC and Protobuf. We also presented a proof-of-concept implementation of our architecture and a ready-to-use evaluation framework for CBR tasks. Finally, we introduced a set of supplementary resources that we believe are useful to other researchers and developers in the domain of CA. The architecture is the culmination of our work over the past years, and we hope to contribute to the standardization of argumentation

[15] https://github.com/recap-utr/arguemapper.

machines. We also await feedback from the community to improve our architecture and resources—everything is hosted on GitHub and open to any kind of contribution.

Software is never a finished product, so there are many potential avenues for future work. First, we will implement the remaining services mentioned in Sect. 4 so that the evaluation framework is no longer restricted to CBR. Subsequently, we will add more services to our architecture to support more tasks in CA. To appeal to a wider audience, we plan to develop an intuitive UI that allows both lay persons and experts to use these services. Another starting point for future work is the creation of a low-code solution for defining new services on the fly—enabling the fast adoption of new ideas and trends in the field CA.

Acknowledgements. This work has been funded by the Deutsche Forschungsgemeinschaft (DFG) within the projects *ReCAP* and *ReCAP-II* (№ 375342983, 2018–2024) as part of the priority program *RATIO* (Robust Argumentation Machines, SPP-1999) as well as the *Studienstiftung*.

References

1. WordNet. An Electronic Lexical Database (1998)
2. Aamodt, A., Plaza, E.: Case-based reasoning: foundational issues, methodological variations, and system approaches. AI Communications **7**(1), 39–59 (1994)
3. Ajjour, Y., et al.: Data acquisition for argument search: the args.me corpus. In: Benzmüller, C., Stuckenschmidt, H. (eds.) KI 2019. LNCS (LNAI), vol. 11793, pp. 48–59. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-30179-8_4
4. Al-Debagy, O., Martinek, P.: A comparative review of microservices and monolithic architectures. In: 2018 IEEE 18th International Symposium on Computational Intelligence and Informatics (CINTI), pp. 000149–000154 (2018)
5. Allen, J.F.: Natural language processing. In: Encyclopedia of Computer Science, pp. 1218–1222 (2003)
6. Bergmann, R., et al.: The ReCAP Project. In: Datenbank Spektrum, pp. 93–98 (2020)
7. Bergmann, R., Lenz, M., Ollinger, S., Pfister, M.: Similarity measures for case-based retrieval of natural language argument graphs in argumentation machines. In: Proceedings of the Thirty-Second International Florida Artificial Intelligence Research Society Conference, pp. 329–334 (2019)
8. Bex, F., Lawrence, J., Snaith, M., Reed, C.: Implementing the argument web. Commun. ACM **56**(10), 66–73 (2013)
9. Bex, F., Snaith, M., Lawrence, J., Reed, C.: ArguBlogging: an application for the Argument Web. J. Web Semant. **25**, 9–15 (2014)
10. Biertz, M., Dumani, L., Nilles, M., Metzler, B., Schenkel, R.: QualiAssistant: extracting qualia structures from texts. In: Proceedings of the 9th Workshop on Argument Mining, pp. 199–208 (2022)
11. Bondarenko, A., et al.: Overview of touché 2023: argument and causal retrieval: extended abstract. In: Kamps, J., et al. (eds.) ECIR 2023, pp. 527–535. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-28241-6_61
12. Britner, S., Dumani, L., Schenkel, R.: AQUAPLANE: the argument quality explainer app. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, pp. 5015–5020 (2023)

13. Chesñevar, C.I., et al.: Towards an argument interchange format. Knowl. Eng. Rev. **21**(4), 293–316 (2006)
14. Dumani, L., Wiesenfeldt, T., Schenkel, R.: Fine and coarse granular argument classification before clustering. In: Proceedings of the 30th ACM International Conference on Information and Knowledge Management, pp. 422–432 (2021)
15. Eden, L., Kantor, Y., Orbach, M., Katz, Y., Slonim, N., Bar-Haim, R.: Welcome to the real world: efficient, incremental and scalable key point analysis. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track, pp. 483–491 (2023)
16. Fröbe, M., et al.: Continuous integration for reproducible shared tasks with TIRA.io. In: Kamps, J., et al. (eds.) ECIR 2023, pp. 236–241. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-28241-6_20
17. Heinisch, P., Orlikowski, M., Romberg, J., Cimiano, P.: Architectural sweet spots for modeling human label variation by the example of argument quality: it's best to relate perspectives! In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, pp. 11138–11154 (2023)
18. Jamshidi, P., Pahl, C., Mendonça, N.C., Lewis, J., Tilkov, S.: Microservices: the journey so far and challenges ahead. IEEE Software **35**(3), 24–35 (2018)
19. Janier, M., Lawrence, J., Reed, C.: OVA+: an argument analysis interface. In: Computational Models of Argument - Proceedings of COMMA 2014, Atholl Palace Hotel, Scottish Highlands, 9–12 September 2014, pp. 463–464 (2014)
20. Lawrence, J., Reed, C.: Argument mining: a survey. Comput. Linguist. **45**(4), 765–818 (2020)
21. Lenz, M., Bergmann, R.: User-centric argument mining with ArgueMapper and Arguebuf. In: Computational Models of Argument, pp. 367–368 (2022)
22. Lenz, M., Bergmann, R.: Case-based adaptation of argument graphs with WordNet and large language models. In: Case-Based Reasoning Research and Development, pp. 263–278 (2023)
23. Lenz, M., et al.: Towards an argument mining pipeline transforming texts to argument graphs. In: Proceedings of the 8th International Conference on Computational Models of Argument, pp. 263–270 (2020)
24. Miller, G.A., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: WordNet: an on-line lexical database. Int. J. Lexicograph. **3**(4), 235–244 (1990)
25. Montani, I., et al.: spaCy: industrial-strength natural language processing (NLP) in python. Zenodo (2023)
26. Nilles, M., Dumani, L., Metzler, B., Schenkel, R.: Trust me, I am an expert: predicting the credibility of experts for statements. In: Proceedings of the Workshops at the 31st International Conference on Case-Based Reasoning (ICCBR-WS 2023), pp. 114–128 (2023)
27. Nilles, M., Dumani, L., Schenkel, R.: QuARk: a GUI for quality-aware ranking of arguments. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2546–2549 (2021)
28. Peldszus, A., Stede, M.: From argument diagrams to argumentation mining in texts - a survey. Int. J. Cognit. Inf. Nat. Intell. **7**(1), 1–31 (2013)
29. Reed, C., Long, D., Fox, M.: An architecture for argumentative dialogue planning. In: Gabbay, D.M., Ohlbach, H.J. (eds.) FAPR 1996. LNCS, vol. 1085, pp. 555–566. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-61313-7_100
30. Reed, C., Norman, T.J.: A roadmap of research in argument and computation. In: Reed, C., Norman, T.J. (eds.) Argumentation Machines, pp. 1–13. Springer, Dordrecht (2004). https://doi.org/10.1007/978-94-017-0431-1_1

31. Reed, C., Walton, D.: Applications of Argumentation Schemes. OSSA Conference Archive (2001)
32. Reed, C., Wells, S., Devereux, J., Rowe, G.: AIF+: dialogue in the argument interchange format. In: Computational Models of Argument: Proceedings of COMMA 2008, pp. 311–323 (2008)
33. Richter, M.M., Weber, R.O.: Case-Based Reasoning: A Textbook. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40167-1
34. Romberg, J.: Is your perspective also my perspective? Enriching prediction with subjectivity. In: Proceedings of the 9th Workshop on Argument Mining, pp. 115–125 (2022)
35. Rücklé, A., Eger, S., Peyrard, M., Gurevych, I.: Concatenated power mean word embeddings as universal cross-lingual sentence representations. arXiv preprint arXiv:1803.01400 [cs] (2018)
36. Slonim, N., et al.: An autonomous debating system. Nature **591**(7850), 379–384 (2021)
37. Stab, C., et al.: ArgumenText: searching for arguments in heterogeneous sources. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 21–25 (2018)
38. Wachsmuth, H., et al.: Argumentation quality assessment: theory vs. practice. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 250–255 (2017)
39. Wachsmuth, H., et al.: Building an argument search engine for the web. In: Proceedings of the 4th Workshop on Argument Mining, pp. 49–59 (2017)
40. Walton, D., Macagno, F.: A classification system for argumentation schemes. Argument Comput. **6**(3), 219–245 (2015)
41. Walton, D., Reed, C., Macagno, F.: Argumentation Schemes (2008)
42. Zhelezniak, V., Savkov, A., Shen, A., Moramarco, F., Flann, J., Hammerla, N.Y.: Don't settle for average, go for the max: fuzzy sets and max-pooled word vectors. arXiv preprint arXiv:1904.13264 [cs] (2019)

Author Index

P. Cimiano et al. (Eds.): RATIO 2024, LNAI 14638, pp. 371–372, 2024.
https://doi.org/10.1007/978-3-031-63536-6