Leipzig University
Institute of Computer Science
Degree Programme Computer Science, B.Sc.

Bootstrapping Training Data for Sentence-Level Trigger Detection

Bachelor's Thesis

Jennifer Rakete

1. Referee: Jun.-Prof. Dr. Martin Potthast

Submission date: November 1, 2023

Declaration

Unless otherwise indicated in the text or references, this thesis is entirely the product of my own scholarly work.

Weißenfels, November 1, 2023

. . . . . .
Jennifer Rakete

Abstract

In recent years, there has been a growing recognition of the need to create inclusive and accessible content that takes into account people with varying sensitivities and triggers. The main goal of this thesis is to bootstrap training data for the development of a classifier to locate and identify triggering concepts (e.g. *Violence*, *Death*) in textual content on a sentence level. This is done to empower readers by preparing them for potentially triggering content by warning them about it. This enables readers to make well-informed decisions about their emotional and mental well-being before engaging with a text. The thesis makes use of a large dataset that includes diverse texts from various genres and topics that were extracted from Archive of Our Own (AO3), a popular online platform that serves as a digital repository for user-generated creative works. The trigger warning set used was created by Wiegmann et al. [2023], consisting of 36 categories of triggering concepts. The methodology entails manually annotating sentences containing triggering terms related to eight distinct potentially triggering concepts. These terms are generated by OpenAI [2023]'s ChatGPT 3.5, a large language model. We conducted 12 experiments on the created dataset of 4,135 sentences and found that our method of sentence retrieval is effective for identifying a large number of positive examples. However, we came to the conclusion that there is no one-size-fits-all model approach for assigning trigger warnings and that a combination of multiple models is more promising.
The best performing model achieved a mean F1 score of 0.51.

Contents

Chapter 1

Introduction

The proliferation of digital content and online platforms has raised concerns about the potential impact on an individual's emotional and mental well-being, particularly when encountering triggering and/or distressing content. Triggering concepts, such as racism and violence, can significantly affect individuals with specific sensitivities or past traumatic experiences. In order to lessen these negative effects, Authors frequently include trigger warnings in their texts to alert readers to the possibility of potentially upsetting material. While these warnings are a valuable tool for promoting emotional well-being, their use is far from universal, with a significant portion of documents lacking such warnings. This deficiency poses an acute problem as readers are left vulnerable to unexpected triggers. Furthermore, the conventional approach of assigning trigger and content warnings to an entire text leaves out information regarding the precise location of potentially triggering concepts within the document. This information is crucial as *Explainability* has grown in importance within AI for the sake of being able to justify the results and judgements of a ML application according to Mishra et al. [2019]. However, in order to detect these potential sites of interest inside a text document, an annotated collection of documents highlighting them is required. Manually annotating these sites of interest in a text is particularly costly since they are scarce and infrequent in comparison to the inconsequential sections. Recognising the significance of this issue, this thesis aims to effectively bootstrap a training dataset to enable a classifier to automatically identify and localise triggering concepts within text. The cost of manual annotation can be reduced by using this strategy, allowing for thorough warning coverage. As a result, the reader can be warned or prepared for potentially triggering content, allowing them to make well-informed decisions about their emotional and mental well-being. Trigger warnings in educational and social contexts aim to create a more inclusive space. The aforementioned fact that trigger warnings are not universal emphasises the need for a univer-

sal taxonomy or set of guidelines for their implementation. A standardised, empirically supported framework can increase their utility by providing a consistent and effective way to serve their intended purpose while also making them an effective asset for NLP tasks like the one presented in this work. One such attempt to standardise trigger warnings has been made by Wiegmann et al. [2023] which we will employ here. The data for this project was obtained from "Archive Of Our Own" (AO3) , a well-known online platform known for its large collection of user-generated content, particularly fan fiction and other creative works. The primary goal of this thesis is to assess the performance and efficacy of the proposed methodology in training the triggering concept classifier. Using the AO3 data, the trigger warning set, and a combination of manual annotation and AI-generated terms, the project aims to create a training dataset that produces a classifier that can accurately identify and localise triggering concepts within text. First, comprehensive lists of trigger terms, consisting of words and phrases commonly associated with triggering content, is compiled using ChatGPT. These lists are carefully curated to represent the respective triggering concepts. Second, a large set of sentences of diverse genres and authors is gathered by searching for documents in the AO3 corpus with frequent occurrences of the generated terms and selecting promising sentences from those documents. Thirdly, those sentences were then annotated manually by three different annotators to create a dataset. The sentences, which contain the identified trigger terms in a harmful context, are treated as positive examples. These positive examples, representing instances of triggering concepts, are used to train the classifier to recognise and localise similar instances in new texts. In the following chapters, we build on the foundation laid out above and contribute to the field of identifying and localising potentially harmful text content. Chapter 2 provides an extensive review of related work in this field, examining previous research and approaches used to address similar challenges. This chapter provides useful insights and contextualises the current study within the broader academic landscape. The methodology used in this project is discussed in depth in Chapter 3. It describes the process of generating trigger terms and contributes a novel method for extracting key terms from ChatGPT. It also provides a systematic approach to filtering examples at the sentence level, effectively filtering for potentially positive examples of triggering concepts. This chapter also highlights the sec:annotationannotation process, which contributes a training dataset consisting of positive examples representing instances of triggering concepts. Chapter 3 demonstrates the feasibility of developing a trigger detection training dataset, laying the groundwork for the development of a classifier capable of identifying a wide range of triggering concepts. Chapter 4 focuses on the experiments conducted and the results obtained. It offers a thorough evaluation of how well the classifier performed

in locating and identifying triggering concepts in texts. The chapter provides empirical support and sheds light on the viability of the suggested strategy. On both a support vector machine (SVM) and a fine-tuned, pretrained BERT model RoBERTa, we ran 12 experiments with two settings (In-Distribution and Out-Of-Distribution) and three setups each. The fine-tuned RoBERTa model, which achieved a mean F_1 of 0.51 with the In-Distribution setting and Extended-Binary setup, was the most effective. In Chapter 5, a thorough analysis of the project's findings, limitations, and implications is conducted. It highlights the strengths and weaknesses of the resulting classifier and discusses potential areas for improvement. Chapter 6 will include a summary as well as the key contributions. The most important of these are that our method of retrieving relevant sentences via keyword lists was found to be effective in identifying a large number of positive examples of triggering concepts, that there is no single model solution for the large variety of triggering concepts, and that a combination of the most effective models per concept is a promising approach.

Chapter 2

Related Work

This chapter delves into the relevant literature and research conducted in the emerging field of triggering content detection and the related fields of sentiment analysis, emotion cause extraction, hate speech detection. First, we overview the fields of sentiment analysis and emotion cause extraction, which focus on extracting sentiments and potential causes for emotions in text, respectively. Next, we review the field of hate speech detection, which involves identifying and mitigating hateful or offensive language within texts. This field emphasises the development of models and algorithms that can automatically detect and classify hate speech, thereby facilitating the creation of safer online environments and fostering inclusive communication. Lastly, we survey the new domain of triggering content detection, which this project is contributing to.

2.1 Sentiment Analysis

Sentiment analysis, also known as opinion mining, is a NLP subdiscipline focusing on classifying, extracting, and evaluating subjective information from text [Liu, 2012]. In the context of text analysis, sentiment analysis and trigger detection are closely related. While sentiment analysis categorises emotional polarity, trigger detection identifies concepts or events that elicit emotional responses. This involves the computational analysis of sentiments, emotions, attitudes, and opinions expressed within the text. Sentiment analysis finds applications in domains like social media analysis, feedback analysis, and market research [Liu, 2012]. Early sentiment analysis categorised polarity, classifying text into positive, negative, or neutral based on sentiment [Kaity and Balakrishnan, 2020]. By calculating the semantic orientation of a given phrase by comparing its similarity to a positive reference word like "excellent" with its similarity to a negative reference word like "poor," Turney [2002] presented a method for categorising reviews as either recommended (positive) or not rec-

ommended (negative). By measuring how closely a given phrase resembled these reference words, Turney determined its sentiment orientation as either positive or negative. This approach allowed for the categorisation of reviews as positive or negative based on the sentiment expressed. They used 410 reviews from a popular opinion platform as their corpus, which included four different review domains, including "Automobiles", "Movies", "Travel Destinations", and "Banks", with an average phrase count of 26.00. Out of these, 170 (41%) were not recommended, while 240 (59%) were. The categorisation accuracy averaged 74.39% across all four domains. This study illustrated the potential of linguistic patterns and feature-based approaches, such as the extraction of phrases containing adjectives or adverbs, as these are good indicators of subjective, evaluative sentences, for the analysis of sentiment in textual data. In trigger detection, it's crucial to recognise specific words and phrases that serve as triggers for particular situations, behaviours, or emotions. The approaches to feature extraction and semantic analysis discussed in the preceding paper, as well as their efficacy, are consequently exceedingly relevant to us for extracting important features or triggers for text data.

In recent approaches, researchers have looked into more nuanced facets of sentiment analysis, such as aspect-based sentiment analysis. Liu [2012] proposed a fine-grained sentiment analysis framework associating sentiments with specific text aspects and entities. For instance, in "*The voice quality of this phone is amazing*," the aspect is "*voice quality*" of the entity "*this phone.*" In the result, the aspect GENERAL is used to represent the entity itself. Attitudes expressed towards specific aspects of a product, service, or subject matter can be more effectively evaluated using this approach. The focus of trigger detection is on identifying specific triggers or events mentioned in text. Trigger detection focuses on identifying specific triggers or text-mentioned events. Both tasks require recognizing elements, comprehending context, and understanding where aspects or triggers appear. In aspect-based sentiment analysis, context aids sentiment polarity determination. Similarly, trigger detection necessitates understanding context for accurate event identification. A racist slur's context impacts whether it's triggering, e.g., educational vs. conversational use. Deep learning sentiment analysis techniques, such as convolutional neural networks (CNNs) or recurrent neural networks (RNNs), have also produced promising results. Kim [2014] trained a straightforward CNN using a single layer of convolution on top of publicly available[1] word vectors that were trained using 100 billion words from Google News. Their model was evaluated on various benchmarks: "MR" for one-sentence movie reviews (identifying positive/negative reviews), "CR" for customer reviews (predicting positive/negative reviews), and

[1] `https://code.google.com/p/word2vec/`

"Subj" for subjectivity (determining subjective/objective sentences). With a vocabulary size range of 5340 to 21323, the average sentence length between the various benchmarks is between 3 and 23. Despite little tuning, this simple model achieved excellent results, which suggests that the pre-trained vectors are 'universal' feature extractors that could be utilised for various classification tasks like, e.g., trigger detection. Learning task-specific vectors through fine-tuning results in further improvements. These results are highly relevant to us in the context of feature extraction and word embeddings. Identifying relevant words or patterns that act as triggers is essential in trigger detection. The feature extraction abilities of CNNs may be used to help identify significant cues that point to the presence of triggers. Further, the paper discusses the use of word embeddings as input to the CNN. Pretrained word embeddings capture semantic relationships between words, which can improve the model's understanding of context. Taking into account the semantic associations of trigger words, word embeddings may enhance trigger detection.

2.2 Emotion Cause Extraction

Yadollahi et al. [2017] separated sentiment analysis into two categories: opinion analysis and emotion analysis. While opinion analysis examines the user's attitude and deals with the expression of opinion, emotion analysis focuses on the identification, categorization, and assessment of the writer's emotions in relation to an event, text, or speech [Khunteta and Singh, 2021]. However, deeper level information regarding emotions, such as the emotion's experiencer, cause, and consequence, needs to be retrieved and analysed [Lee et al., 2010]. Initially, ECE techniques focused on rule-based systems and manual annotation. Lee et al. [2010] presented a linguistic-driven rule-based system for emotion cause recognition that detected emotion causes in text using language cues such as causative verbs ("to cause"), action verbs ("to think about"), epistemic markers ("to see"), conjunctions ("because"), and prepositions ("for"). Based on the list of 91 Chinese major emotion keywords defined by Chen et al. [2009], they extracted 6,058 phrases via keyword matching from the Sinica Corpus, a tagged balanced Mandarin Chinese corpus. Each instance includes the focus sentence with the emotion key word `<FocusSentence>`, as well as the sentences that come before and after it (the `<PrefixSentence>` and `<SuffixSentence>`, respectively). Following that, they compute the distribution of cause event types as well as the position of cause events in relation to emotion keywords and experiencers. Finally, using the aforementioned language cues, they form 15 linguistic rules for identifying the cause of the corresponding emotion verb. They evaluated their approach using a two-phase

performance evaluation scheme and reported promising results for cause occurrence detection in addition to cause event detection. The study demonstrates the effectiveness of a rule-based approach in identifying the causes or triggers of specific emotions, emphasising the potential of linguistic patterns in capturing causal information. Similarly, rule-based approaches to identifying specific triggers in text could benefit trigger detection. The rules and patterns developed for emotion cause detection tasks could potentially be adapted or extended for trigger detection tasks.
In their paper "Extracting Causes of Emotions from Text," Neviarouskaya and Aono [2013] describe a technique for automatically identifying the linguistic relationships between an emotion and its cause as well as the extraction of the phrases describing the causes of the emotions. To accomplish this, they created a corpus of 532 sentences containing approximately 130 tokens (emotion words) distributed across 22 emotion types, which they manually annotated. 118 emotion tokens were found to be productive, with each emotion token resulting in at least one cause-containing sentence. For example, '*glad*' and '*happy*' are associated with the *Joy* emotion class; '*scared*' and '*terrified*' are associated with the *Fear* emotion class; and '*awe*' and '*esteem*' describe the *Admiration* emotion class. Their method for determining emotion causes is based on an examination of syntactic and dependency information from the parser. They apply Connexor Machine Syntax[2] to each sentence to obtain lemmas, dependencies, syntactic, and morphological information. Using the parser output, the algorithm then detects and extracts phrases that characterise the emotion caused by prepositions. After analysing the errors, they were able to improve their technique, resulting in a 15% increase in the accuracy of their proposed method and very good overall results. The paper employs semantic analysis to comprehend the relationships between emotions and their causes. This form of analysis is also greatly relevant to trigger detection for understanding the semantic associations between triggers and events.

2.3 Hate Speech Detection

According to the definition given by Levy et al. [2000], hate speech is any speech that disparages an individual or a group based on one or more of their racial, ethnic, gendered, sexual, national, religious, or other characteristics. Hate speech detection, a critical task in natural language processing, is similar to trigger detection in that both involve the identification of specific linguistic cues and context for meaningful analysis. In their research, Waseem and Hovy [2016] examine the automatic detection of hate speech on social media. The

[2] `https://www.connexor.com/nlplib/?q=msyn`

study looks into whether identifying specific words and symbols or profiling users' behaviours and backgrounds is more effective in detecting hate speech. To conduct the research, the authors created a manually annotated dataset composed of tweets that either contained hate speech (limited to sexist or racist content) or were non-hateful. The dataset contained 16,914 annotated tweets gathered through manual keyword searches and subsequent hashtag queries related to hate speech. The authors used a variety of linguistic and social cues to distinguish hate speech. Linguistic characteristics included the most frequently occurring tokens, tweet length, and n-grammes. Gender and geographical location were examples of social features. The study discovered that using character n-grammes of up to length 4, along with gender as an additional feature to provide context, produced the best results. The findings of this study are relevant to trigger detection because they emphasise the importance of analysing not only specific linguistic triggers but also contextual cues and user behaviour. The focus on user-related features aligns with trigger detection's emphasis on understanding the context in which certain events, actions, or emotions are initiated. This study's insights underscore the importance of considering broader contextual elements beyond isolated words, reinforcing the idea that triggers are intricately linked with their surroundings and contextual cues.

To distinguish between hate speech and offensive language, Davidson et al. [2017] trained a model to categorise tweets into three categories: hate speech, offensive language, or neither, and then analysed the findings to better understand how to discern between them. They started with a hate speech lexicon, which contained words and phrases identified as hate speech by internet users. Using the Twitter API, they obtained a sample of tweets from 33,458 Twitter users containing the terms from the lexicon. They then extracted and sampled each user's timeline, yielding a corpus of 25K tweets containing hate speech lexicon terms that were then manually annotated. Each tweet was then lowercased and stemmed to generate unigram, bigram, and trigram features based on its tf-idf. To capture information about syntactic structure, they use NLTK to build Penn Part-of-Speech (POS) tag unigrams, bigrams, and trigrams, also known as grammatical tagging. POS is the automatic assignment of part-of-speech tags to words in a sentence [Chiche and Yitagesu, 2022]. They also assign sentiment scores to each tweet by using a sentiment lexicon. Davidson et al. [2017] also include binary and count indications for hashtags, mentions, retweets, and URLs, as well as characteristics for the quantity of characters, words, and syllables in each tweet. Their model was a logistic regression with L2 regularisation, allowing them to investigate the predicted probabilities of class membership. The final model was trained on the entire dataset using the one-versus-rest framework and then predicted the label for each tweet. The

findings of this study show that while lexical methods are effective at identifying potentially offensive terms, they are ineffective at identifying hate speech. The findings of this study are relevant to trigger detection because they highlight the complexities of distinguishing between related yet distinct linguistic concepts, which is similar to identifying triggers in a larger context. The difficulty in distinguishing between hate speech and offensive language mirrors the difficulty in recognising specific linguistic cues that initiate specific reactions in trigger detection while taking the surrounding context into account. The nuanced understanding required to distinguish between these categories reflects the complexities inherent in identifying linguistic triggers that initiate specific emotions, actions, or events.

2.4 Triggering Content Detection

The field of triggering content detection is relatively new, introduced by Wolska et al. (2022). Since this thesis is a contribution to this field, we will focus on the differences between the previously discussed works and this one rather than placing the research in the context of triggering content detection. Various forms of media, regardless of their nature, can contain distressing content that has the potential to elicit uncomfortable emotional responses, especially among individuals with sensitivities or past traumatic experiences. To address this concern, content creators have begun to include "trigger warnings" before their work to caution consumers about potentially disturbing material. These warnings, typically conveyed through keywords or phrases, are inserted manually, leading to some creators omitting them due to a lack of awareness or inclination. Automating this process could ensure widespread trigger warnings without the need for manual intervention from creators. Addressing this research gap, Wolska et al., 2022 introduced the task of automating trigger detection, specifically focusing on violence triggers. Their contribution is threefold: the aforementioned introduction of the new task of automated trigger detection; the creation of a corpus sourced from Archive of Our Own (AO3); and the development and evaluation of a violence trigger detection model. AO3 offers four predefined content warnings: Major Character Death, Underage, Rape/Non-Con, and Graphic Depictions of Violence. Authors can also include additional freeform tags, such as "romance" or "monsters." The corpus was created by crawling the entire AO3 anthology, yielding 7,866,512 works, 571,525 of which were labelled Graphic Depictions of Violence. Additionally, for each work labelled as Violence, they calculated the proportion of its other tags that also appear in other works labelled as Violence and created two sub-corpora with two thresholds of 50% of other tags appearing in other

Violence works and 40% of other tags appearing in other Violence works. Negative examples were extracted from works labelled No Archive Warnings Apply. For the model baseline, they employed a support vector machine (SVM). The reserachers went on to use a pretrained BERT transformer model with 12 layers and 110M parameters. The outcomes highlight the non-triviality of the task while also demonstrating an efficient classification. However, this work diverges from the present thesis as it primarily focuses on a single trigger warning label, in contrast to the broader range of trigger warnings considered in this study. While Wolska et al. [2022] introduced and showed the viability of automated trigger detection, this thesis expands on their findings by broadening the scope of the detection model to include a wide range of trigger warnings. Another pioneering study in the realm of automatic trigger detection is the work conducted by Stratta et al. [2020], which presents a proof-of-concept for generating warnings on the client side through keyword identification and sentiment analysis. Their tool, the DeText browser extension for Google Chrome, was tested on websites with and without sensitive material about sexual violence. The system works by extracting HTML source code from a webpage, passing it to a Python server, and extracting visible text from the source code using the `<p>` and `<div>` HTML tags. This extracted text then undergoes analysis. The algorithm comprises two main steps. The initial step involves a keyword search utilising an explicit and implicit keyword list. Explicit keywords are directly linked to sexual violence (e.g., “rape”, “sexual assault”, etc.), while implicit keywords have a broader contextual usage (e.g., “pain”, “force”, etc.), with the final list containing over 200 implicit and explicit keywords. The algorithm searches each paragraph using this keyword list and flags it if it contains at least one word from the explicit list and at least two words from the implicit list. The second step entails sentiment analysis, determining polarity within flagged paragraphs. A statement is labelled "polarised" if its magnitude exceeds 0.05, whereas neutral statements remain unflagged due to their factual nature. The system constructs a data structure for each marked paragraph, storing explicit and implicit phrase counts along with sentiment polarity. Using these data structures, the system then computes a final value and determines whether or not to display a content warning. Stratta et al. [2020] tested their system against known-classified web pages and found it to be highly accurate. Despite demonstrating the feasibility of automatic trigger detection, this proof-of-concept is limited to a single trigger warning, in contrast to the present thesis, which aims to broaden the spectrum of identifiable triggers. Furthermore, this thesis intends to achieve sentence-level localization, a departure from the conventional approach of assigning trigger warnings to entire texts.

Chapter 3

Bootstrapping of Examples

The methodology used in this thesis will be covered in this chapter. First, we review the acquisition of the source data and structuring of the employed standardised trigger warning set by Wiegmann et al. [2023]. Second, we overview the generation of trigger term lists with ChatGPT 3.5. Following that, we will go over how to extract relevant examples from previously obtained data. Finally, we take a look at the annotation process.

3.1 Source Data

Archive of Our Own (AO3)[1] is a non-profit online repository that hosts fan fiction works submitted by users in a variety of genres. It is a hub for diverse creative expressions, with a vast range of content that includes stories, poems, and art. The corpus used in this thesis was generated by crawling the full anthology of works available on AO3, yielding a large database of 7.8 million works. These works were then divided into individual chapter chunks to allow for more granular analysis.

To create a standardised set of trigger warnings, Wiegmann et al. [2023] manually processed trigger warning guidelines from eight major English-speaking universities. The taxonomy includes 36 labels divided into 7 supercategories with 29 subcategories. The subcategories are narrower in definition and have clear semantics, making them ideal for classification tasks. These might include *Child Abuse*, *Transphobia*, or *Graphic Violence*, where the meaning is explicit and specific to an explicit type of content. The supercategories, on the other hand, are more general categories that cover a broader range of potential triggers. These might include *Sexual*, *Discrimination*, or *Aggression*. The

[1] `https://archiveofourown.org/`

supercategories serve as umbrella categories for multiple subcategories. For instance, the supercategory *Sexual* might encompass subcategories like *Incest* and *Pornography*. Four key observations about this taxonomy are highlighted in the study. First, the categories' granularity varies, with some being highly specific (e.g., *Child Abuse*) and others being more general (e.g., *Abuse*).
Second, the subcategories frequently form natural clusters, such as different types of *Discrimination*, which can be grouped under a supercategory. Third, the taxonomy is not exhaustive due to the complex and open-ended nature of triggers, necessitating the inclusion of the 7 supercategories. Fourth, the initial lexical and semantic definitions of these labels were insufficient for accurate document annotation, necessitating additional refinement to make them more precise. The study concludes that this taxonomy is not only useful for creating more effective and standardised trigger warnings in human communication, but it also has significant potential for improving NLP tasks such as content filtering and sentiment analysis by using a unified set of labels. We will use this taxonomy in the following steps to bootstrap a training set for sentence-level trigger detection.
However, considering all 36 categories would be beyond the scope of this thesis due to the high cost of the annotation process, we focused on 8 subcategories, half of which were from the supercategory *Aggression* (*Death*, *Violence*, *Abduction*, *War*) and the other half from the supercategory *Discrimination* (*Misogyny*, *Racism*, *Homophobia*, *Ableism*).

3.2 Generation of Trigger-Term Lists

We worked with ChatGPT 3.5 [OpenAI, 2023], an OpenAI language model, to create a list of trigger terms. For each of the subcategories we prompted ChatGPT with the same prompt for consistency except for some syntactic adaptations to for each individual warning. For example, to generate terms related to the *War* subcategory, we framed our prompt as follows:
"*Now give me a list of verbs relating to war that may be triggering to people so I can avoid them.*"
The results were manually reviewed and the mean amount of phrases per cleaned up list was 33.875. ChatGPT is a conversational agent created by OpenAI that is based on the GPT (Generative Pre-trained Transformer) architecture. We used ChatGPT 3.5 for this study, which is one of the later iterations of this language model. As of its most recent update, GPT-3.5 had 175 billion machine learning parameters, making it one of the most advanced publicly available language models. These parameters are tuned to generate human-like text in response to a wide range of prompts, making the model

extremely versatile in handling a wide range of natural language processing tasks, from text generation to summarization and translation. Despite the model's flexibility, it has some "guardrails" built in to prevent the production of sensitive or harmful content. These safeguards include restrictions on, among other things, generating hate speech, promoting violence, and sharing graphic content. To generate a comprehensive list of trigger terms corresponding to our subcategories, we had to carefully engineer our prompts as it was critical to effectively circumventing ChatGPT's guardrails while ensuring that the model understood our ethical intent in collecting such potentially sensitive terms. The explicit statement about our intention to use these terms to create more inclusive and respectful spaces helped the language model comply with our request. The term lists generated by ChatGPT 3.5 underwent an optimisation process for enhanced precision. We manually pruned terms that were either too broad, lacked specificity or were redundant in their correlation with the trigger warning subcategories during this phase. For example, terms like "hit" for the *Violence* subcategory and "force" for the *Sexual Abuse* subcategory were excluded. Table 1 shows the numbers for the original and cleaned lists for each trigger warning subcategory. These cleaned up terms serve as the foundation for the next phase of our project. Specifically, the terms are used to query the Archive of Our Own (AO3) corpus to retrieve sentences that can be positively associated with each trigger warning subcategory. Table 3.1 shows the list-building procedure on the example of the *Misogyny* keyword and phrase list. The list of keywords and phrases was split in the middle according to the order provided by ChatGPT. Depending on which split a keywod or phrase belonged to, the sentence containing the keyword or phrase was assigned either a set ID of 1 or 2. These IDs are utilised later for testing generalisation capabilities in the Experiments and Results chapter. Manual adjustments were made in the distribution of these keywords and phrases across the two halves for certain trigger warning subcategories to ensure a high number of results for both sets.

Table 3.1: The original keyword and phrase list of *Misogyny* with the cleaned-up phrases as well as the split for the two sets (continued on the next page).

Trigger	Misogyny
Prompt	Give me a list of misogynistic language that must be avoided in a story for people that are triggered by it.
Phrases	bitch, slut, whore, hoe, skank, dragon lady, bimbo, femme fatale, hysterical, nagging, btch, whre, cunt, gold digger, dumb blonde, bossy, too emotional, shrill, crazy, drama queen, attention seeker, baby-making machine, trophy wife, mail-order bride, maid, barefoot and pregnant, you're too pretty to be, asking for it, boys will be boys, locker room talk, not like other girls, must be that time of the month, too aggressive, hormonal, playing the gender card, oversensitive, man up, like a girl, such a girl, pussy , be a man, ladylike, real women have curves, stop being so hormonal, you're not a real woman if you don't have children, you're too old to be single, you'll change your mind about, you're too pretty to be single, you need a man to take care of you, feminazi, man-hater, girly-man, boy toy, cougar, old maid, tomboy, pimp, ho-bag, testosterone-fueled, chick, daddy issues, emasculate
Cleaned	bitch, slut, whore, hoe, skank, dragon lady, bimbo, femme fatale, hysterical, nagging, btch, whre, cunt, gold digger, dumb blonde, bossy, too emotional, shrill, drama queen, attention seeker, baby-making machine, trophy wife, mail-order bride, barefoot and pregnant, too pretty to be, asking for it, boys will be boys, locker room talk, not like other girls, must be that time of the month, too aggressive, hormonal, playing the gender card, oversensitive, man up, like a girl, such a girl, be a man, ladylike, real women have curves, you're not a real woman if you don't have children, too old to be, you need a man to take care of you, feminazi, man-hater, girly-man, boy toy, cougar, tomboy, pimp, ho-bag, testosterone-fueled, chick, daddy issues, emasculate

Set 1	bitch, slut, whore, hoe, skank, dragon lady, bimbo, femme fatale, hysterical, nagging, btch, whre, cunt, gold digger, dumb blonde, bossy, too emotional, shrill, drama queen, attention seeker, baby-making machine, trophy wife, mail-order bride, barefoot and pregnant, too pretty to be, asking for it, boys will be boys
Set 2	locker room talk, not like other girls, must be that time of the month, too aggressive, hormonal, playing the gender card, oversensitive, man up, like a girl, such a girl, be a man, ladylike, real women have curves, you're not a real woman if you don't have children, too old to be, you need a man to take care of you, feminazi, man-hater, girly-man, boy toy, cougar, tomboy, pimp, ho-bag, testosterone-fueled, chick, daddy issues, emasculate

3.3 Retrieval of Examples

For potentially positive examples we queried the individual chapters of the documents retrieved from AO3 by looking for documents with a high occurrence of the respective trigger terms and phrases. In these documents we then searched specifically for sentences containing the trigger terms or phrases to retrieve for annotation as well as two sentences before and two after the target sentence to provide context for the annotation process. The context is important for the annotators to be able to categorise an example with certainty and, in doubt about the to-be-annotated sentence, refer to the context in order to make a decision. The corpus was hosted on ElasticSearch[2], a full-text search and analytics engine. The actual retrieval process was automated by a Python script that interacted with ElasticSearch's API. This enabled a seamless, programmatic search within the corpus for sentences containing the specified potentially triggering terms as well as a simple removal of HTML tags. As a result of this method, we were able to retrieve highly relevant samples for each trigger warning subcategory. The output of this process was formatted in JSON (JavaScript Object Notation).

[2] `https://www.elastic.co/`

3.4 Annotation

We used LabelStudio [Tkachenko et al., 2020-2023] to annotate sentences from our previously collected corpus. We started the process by importing the JSON-formatted sentences into the LabelStudio platform. Within LabelStudio, we designed a structured and easily readable labelling interface to facilitate accurate and consistent annotation. This interface included the current sentence being evaluated, the preceding sentence for context, and the following sentence for additional context. In addition, we provided three checkboxes for annotators to use when categorising each sentence: 'positive' (indicating it is a valid example of the respective trigger warning subcategory), 'negative' (indicating it is not an example), or 'unclear' (used when categorisation was ambiguous). To further clarify the annotator's rationale for when a sentence's label was deemed unclear, we included a small text field for annotators to provide brief explanations. The annotators then discussed the ambiguous cases

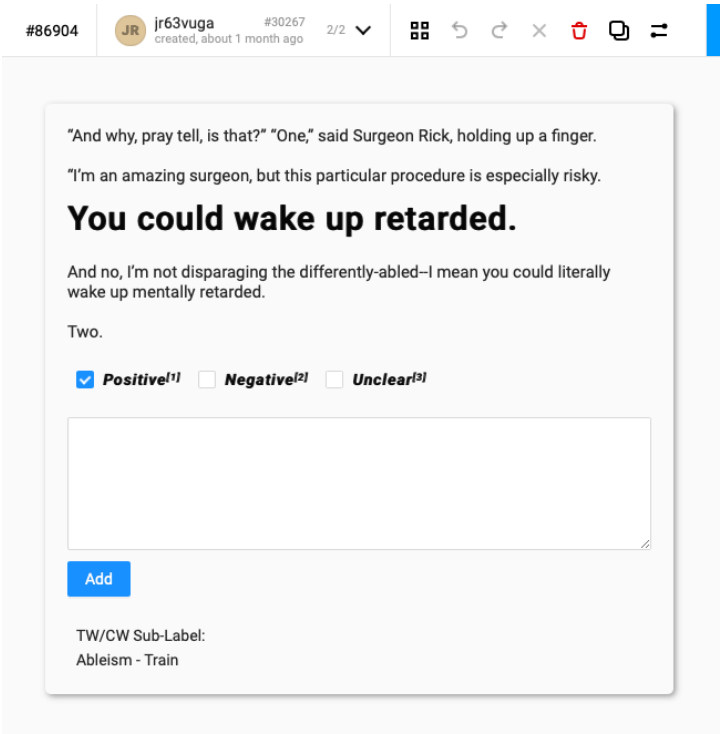

Figure 3.1: Labelling interface on LabelStudio

in order to reach a consensus on whether to annotate an ambiguous sentence as positive or negative. The final categorisation of a sentence was based on the majority of the votes. We implemented an annotator agreement strategy to reduce the potential introduction of personal bias into the datasets. Each

sentence was evaluated independently by a minimum of three annotators. The first annotator labelled sentences until there were at least 50 positive examples for each set for each of the eight trigger warnings, for a total of 100 positive examples per individual trigger warning. The other annotators then labelled all sentences that had been labelled by the first annotator. This was necessitated by the uneven distribution of positive to negative examples among the individual trigger warnings, with some positive-to-negative labelled sentence ratios as low as 1:10. As a result, annotating the same number of sentences for each set would result in either a very low number of positive examples or a very high annotation cost, despite the fact that we are only focusing on 8 subcategories. The results in Table 3.2 reveal that 947 cases out of 4135 have been labelled as positive, accounting for approximately 22.9% of all positive examples in all retrieved, annotated phrases. Given that the majority of sentences in a text are irrelevant or negative examples for our purposes, this percentage should be interpreted as a retrieval success. However, further improvements are possible in order to reduce the annotation load as for example the trigger warning *Violence* contained many negative examples due to very broad keywords and phrases like "hit" and "crush" despite prior pruning. Furthermore, the trigger warning *Death* featured many fantastic concepts associated with it (e.g., necromancy, vampirism), which the annotators mainly labelled as negative due to their lack of basis in reality. The cost of annotation would be reduced if the terms associated with these concepts were removed from the keyword list. This method enabled us to assess annotator consensus and mitigate the impact of individual biases. While it was difficult to completely eliminate bias, this method assisted in making the labelled datasets as objective and accurate as possible. The resulting datasets were then used in our experiments to assess the efficacy of our approach.

Table 3.2: Dataset statistics including the number and length in words of the sentences, number of positive and negative annotations per set ID and number of phrases in the original and cleaned-up keyword lists.

Warning	**Sentences**		**Set 1**		**Set 2**		**Keywords**	
	num.	len.	pos.	neg.	pos.	neg.	orig.	clean
Abduction	511	83	57	195	74	185	38	25
Ableism	255	83	50	112	46	47	38	27
Death	544	83	79	99	35	331	29	28
Homophobia	313	79	54	75	51	133	47	31
Racism	267	90	66	2	52	147	43	37
Misogyny	377	84	60	92	24	201	62	55
Violence	1,041	92	107	568	81	285	20	18
War	827	95	50	152	61	564	170	50
Total	4,135	88	523	1,295	424	1,893	447	271

Chapter 4

Experiments and Results

The process used here to identify trigger warnings in a text is a binary classification task. Given a warning and a sentence, the classifier determines whether or not to assign the trigger warning. The preprocessing of the sentences in the datasets additionally done to the removal of HTML tags at the retrieval stage includes the normalisation of quotes, the removal of leading special characters and parsing relics, the omission of strings shorter than three characters and the lowercasing of all characters. In order to test whether the models can generalise to unknown domains, the experiments are split into two distinct settings, the Out-Of-Distribution and In-Distribution splits. We designed three setups to be used in each split:

- **Strict-One-Class:** One-vs-All, multilabel setup that uses only positive examples from all warnings for training. If we only have a few negative examples but many different classes, this setup is promising.
- **Lenient-One-Class:** One-vs-All setup with two distinct models, each trained with a training set containing all positive examples of either the *Discrimination* or *Aggression* supercategories, as well as all negative examples of the corresponding subcategory and all positive examples of the other supercategory as additional negative examples.
- **Extended-Binary:** This setup trains individual classifiers—eight in total, one for each of the annotated subcategories—that are trained on both in-label positives and negatives, as well as negatives from the other supercategory. This is done to balance the limited training data, which is especially important for warnings with few negative examples. We only expand with negative examples from the other supercategory to avoid expanding with examples that could also be positive for subcategories of

the same supercategory (e.g., examples positive for *Abduction* could also be positive for *Violence*). To classify a warning-sentence pair, we only use the model trained to classify this particular warning.

All of these setups were tested using both settings on a SVM and the pre-trained BERT model RoBERTa [Liu et al., 2019]. We used the "roberta-base" checkpoint[1] and fine-tuned it for fan fiction documents with the `Trainer` routine provided by HuggingFace. The data used were fan fiction documents in English only, sourced from the AO3 corpus. This makes a total of 12 experiments.

4.1 Out-Of-Distribution

For the Out-Of-Distribution split we constructed the training dataset with all sentences retrieved with the keywords from set 1 and the test dataset with all sentences retrieved with the keywords from set 2. As a result, the keyword lists do not overlap, and the model will see new keywords in the test dataset for the warnings that it previously trained on.

SVM

For each of the Strict-One-Class, Lenient-OneClass and Extended-Binary setup we trained an SVM with a sigmoid kernel and a gamma of 1.5. We used the "SVC" Scikit-Learn class in the version 1.3.0. and set the `class_weight` parameter to "balanced". We experimented with different kernels and found the sigmoid kernel to be the most effective. For the Strict-One-Class and Lenient-One-Class setups, we used the Scikit-Learn `OneVsRestClassifier` module to implement our One-vs-All approach [Pedregosa et al., 2011]. For the selected features we used the `TfidfVectorizer` class also from Scikit-Learn.

[1] `https://huggingface.co/roberta-base`

Results SVM

Table 4.1: Results of the SVM experiments with the Out-Of-Distribution split setting.

Warning	**strict-one-class**			**lenient-one-class**			**extended-binary**		
	f_1	prec.	rec.	f_1	prec.	rec.	f_1	prec.	rec.
Abduction	0.29	**0.46**	0.22	0.26	0.48	0.18	0.18	0.29	0.14
Ableism	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Death	0.30	0.32	0.29	**0.36**	**0.50**	0.29	0.19	0.14	**0.31**
Homophobia	0.25	0.33	0.20	0.33	0.40	0.27	0.04	0.33	0.02
Racism	0.07	0.22	0.04	0.07	0.22	0.04	0.03	0.11	0.02
Misogyny	0.27	0.22	0.33	0.20	0.25	0.17	0.06	0.08	0.04
Violence	**0.32**	0.27	0.40	0.33	0.36	0.31	**0.24**	**0.45**	0.16
War	0.29	0.20	**0.51**	0.30	0.20	**0.57**	0.09	0.15	0.07
Mean	0.22	0.25	0.25	0.23	0.30	0.23	0.10	0.19	0.09

The results show that this model and setting is most effective for the concepts of the supercategory *Aggression* with all of the best values highlighted per column belonging to one of the four subcategories of *Aggression*, while the subcategories of *Discrimination*, but especially the *Racism* subcategory, underperform. The effectiveness for the Extended-Binary setup is the lowest for this model with a mean F_1 of 0.10 while the Lenient-One-Class setup is the most effective with a mean F_1 of 0.23. This is also the case for the SVM model in the In-Distribution setting, with the least effective setup being the Extended-Binary setup with a mean F_1 of 0.11 and the most effective being the Lenient-One-Class setup with a mean F_1 of 0.43. However, the model in the In-Distribution setting is a lot more effective for the *Discrimination* subcategories. The fine-tuned RoBERTa model on the other hand is most effective with the Extended-Binary setup and least effective with the Lenient-One-Class setup in both settings with the RoBERTa model in the Out-Of-Distribution setting being more effective by 0.14 - 0.20 than the Out-Of-Distribution SVM model.

RoBERTa

We used the fine-tuned RoBERTa model with the standard parameters of the HuggingFace "Trainer" class `evaluation_strategy` and `save_strategy` set to "epoch", `load_best_model_at_end` set to "True" and `metric_for_best_model`

set to "f1_macro" for each of the presented setups, with an adjusted learning rate of 2e-5 and a weight decay of 0.01. We trained for 5 epochs and used a batch size of 12 for the Strict-One-Class and Lenient-One-Class setups. We trained the Extended-Binary setup for 4 epochs with the same batch size as the other setups.

Results RoBERTa

Table 4.2: Results of the RoBERTa experiments with the Out-Of-Distribution split setting.

Warning	**strict-one-class**			**lenient-one-class**			**extended-binary**		
	f_1	prec.	rec.	f_1	prec.	rec.	f_1	prec.	rec.
Abduction	0.45	0.38	0.54	0.00	0.00	0.00	0.49	0.43	0.57
Ableism	0.25	0.33	0.20	0.14	0.33	0.09	0.18	0.50	0.11
Death	0.33	0.23	0.57	0.46	0.42	**0.51**	0.38	0.27	**0.69**
Homophobia	0.29	0.38	0.24	0.33	0.48	0.25	0.23	0.44	0.16
Racism	0.00	0.00	0.00	0.00	0.00	0.00	0.14	0.18	0.12
Misogyny	0.10	0.08	0.12	0.13	0.14	0.12	0.11	0.15	0.08
Violence	**0.51**	**0.46**	0.57	**0.52**	**0.57**	0.48	**0.57**	**0.52**	0.64
War	0.30	0.19	**0.72**	0.30	0.31	0.30	0.30	0.22	0.48
Mean	0.28	0.26	0.37	0.24	0.28	0.22	0.30	0.34	0.36

The results show that this model is, just as the Out-Of-Distribution SVM model, more effective for the subcategories of *Aggression* with especially the subcategories *Racism* and *Misogyny* underperforming. The least effective setup for this model and setting is the Lenient-One-Class setup with a mean F_1 of 0.24 and the most effective the Extended-Binary setup with a mean F_1 of 0.30. This is similar to the RoBERTa model in the In-Distribution setting with the least effective model being the Strict-One-Class setup with a mean F_1 of 0.41 and the most effective setup being the Extended-Binary setup with a mean F_1 of 0.51, which is marginally more effective than the Lenient-One-Class setup by 0.01. Our observation of the hightened effectiveness of the In-Distribution SVM model for the *Discrimination* subcategories is also true for this model, with the best values highlighted per column in the In-Distribution RoBERTa model pimarily being *Ableism*, *Racism* and *Homophobia*.

4.2 In-Distribution

For the Out-Of-Distribution split we used all of the annotated sentences and sorted the examples by chance into the training or test dataset in a ratio of 80:20. When compared to the previous split, this increases the likelihood that the model has seen the majority of the keywords already during training.

SVM

As in the previous split, we trained an SVM with a sigmoid kernel and a gamma of 1.5 additionally with the `OneVsRestClassifier` for the Strict-One-Class and Lenient-One-Class setups and the same parameters and feature selection as before.

Results SVM

Table 4.3: Results of the SVM experiments with the In-Distribution split setting.

Warning	**strict-one-class**			**lenient-one-class**			**extended-binary**		
	f_1	prec.	rec.	f_1	prec.	rec.	f_1	prec.	rec.
Abduction	0.41	0.30	0.67	0.34	0.27	0.48	0.00	0.00	0.00
Ableism	0.57	0.52	0.64	0.60	0.56	0.64	0.00	0.00	0.00
Death	0.25	0.16	0.64	0.31	0.21	0.57	**0.24**	0.67	**0.14**
Homophobia	0.42	0.29	**0.78**	0.46	0.30	**0.94**	0.19	0.67	0.11
Racism	**0.66**	**0.63**	0.68	**0.71**	**0.71**	0.71	**0.24**	**0.80**	**0.14**
Misogyny	0.29	0.22	0.42	0.33	0.25	0.50	0.13	0.33	0.08
Violence	0.41	0.29	0.67	0.33	0.35	0.31	0.05	0.50	0.03
War	0.31	0.21	0.67	0.33	0.23	0.57	0.00	0.00	0.00
Mean	0.42	0.33	0.65	0.43	0.36	0.59	0.11	0.37	0.06

The results of this model show a better effectiveness for the subcategories of *Discrimination* especially for *Racism* in comparison to the model in the Out-Of-Distribution setting, which showed a better effectiveness for the subcategories of *Aggression*. The least effective setup is, as in the Out-Of-Distribution setting, the Extended-Binary setup with a mean F_1 of 0.11, and the most effective setup the Lenient-One-Class setup with a mean F_1 of 0.43, which is marginally more effective than the Strict-One-Class setup by 0.01.

RoBERTa

As in the previous split, we used the fine-tuned RoBERTa model with the same standard parameters for the HuggingFace `Trainer` class, an adjusted learning rate of 2e-5, and a weight decay of 0.01. All setups are trained on batches of 12, and the Strict-One-Class and Lenient-One-Class setups train for 5 epochs, while the Extended-Binary setup trains for 4 epochs.

Results RoBERTa

Table 4.4: Results of the RoBERta experiments with the In-Distribution split setting.

Warning	**strict-one-class**			**lenient-one-class**			**extended-binary**		
	f_1	prec.	rec.	f_1	prec.	rec.	f_1	prec.	rec.
Abduction	0.45	0.30	0.90	0.52	0.39	0.81	0.41	0.32	0.57
Ableism	**0.62**	**0.58**	0.68	**0.75**	0.66	0.86	0.53	**0.75**	0.41
Death	0.21	0.12	0.86	0.37	0.24	0.79	**0.65**	0.59	**0.71**
Homophobia	0.43	0.28	**0.94**	0.42	0.27	**0.94**	0.49	0.47	0.50
Racism	0.56	**0.58**	0.54	0.71	**0.68**	0.75	**0.65**	0.71	0.61
Misogyny	0.30	0.18	0.83	0.31	0.20	0.67	0.24	0.40	0.17
Violence	0.40	0.27	0.74	0.57	0.65	0.51	**0.65**	0.63	0.67
War	0.32	0.20	0.90	0.36	0.23	0.90	0.48	0.48	0.48
Mean	0.41	0.31	0.80	0.50	0.42	0.78	0.51	0.54	0.52

The results of this model and setting show a better effectiveness for the subcategories of *Discrimination* especially for *Racism*, *Ableism* and *Homophobia* in comparison to the model in the Out-Of-Distribution setting, which showed a better effectiveness for the subcategories of *Aggression*. The least effective setup for this model in this setting is the Strict-One-Class setup with a mean F_1 of 0.41, which is 0.01 - 0.02 less effective than the Strict-One-Class and Lenient-One-Class setups for the SVM model in the same setting. The most effective setup is again the Extended-Binary setup with a mean F_1 of 0.51 which is also the most effective model overall. It is 0.08 more effective than the best SVM model with the Lenient-One-Class setup in the In-Distribution setting.

Chapter 5

Discussion

Figure 5.1 shows the effectiveness of the fine-tuned RoBERTa models for the In-Distribution and Out-Of-Distribution settings. With a mean F_1 of 0.30 in the Out-Of-Distribution setting and 0.51 in the In-Distribution setting, the RoBERTa models trained for the Extended-Binary setup are the most effective. The Extended-Binary setting only slightly outperforms (0.01–0.06) the Lenient-One-Class setup, while the Strict-One-Class setup is only competitive with the Out-Of-Distribution setting with a mean F_1 of 0.28. However, we also observed that it depended on whether the model was trained in the In-Distribution or Out-Of-Distribution setting if the model was more effective for subcategories of *Aggression* or *Discrimination*. Generally, the models in the In-Distribution setting have a much higher effectiveness for *Discrimination* subcategories when compared to the results of the same subcategories in the Out-Of-Distribution setting.

The SVM models, on the other hand, consistently performed best in the Lenient-One-Class setup, with F_1 means of 0.23 with the Out-Of-Distribution setting and 0.43 for the In-Distribution setting. Here, the Lenient-One-Class setup is closely followed by the Strict-One-Class setup, with only a marginally small difference (0.01). In comparison, the Extended-Binary setup performs poorly, with F_1 means of 0.1 and 0.11 for the Out-Of-Distribution and In-Distribution settings, respectively. The poor performance of the Extended-Binary setup for SVM experiments can be explained by the SVM algorithm's general loss of performance when there is a lot of noise in the data, such as when the target classes overlap, as is the case here. The frequent overlap of triggering concepts is a limitation of our research, since it makes defining clear decision boundaries difficult. Triggering concepts are oftentimes interconnected, making it challenging to establish distinct classification categories.

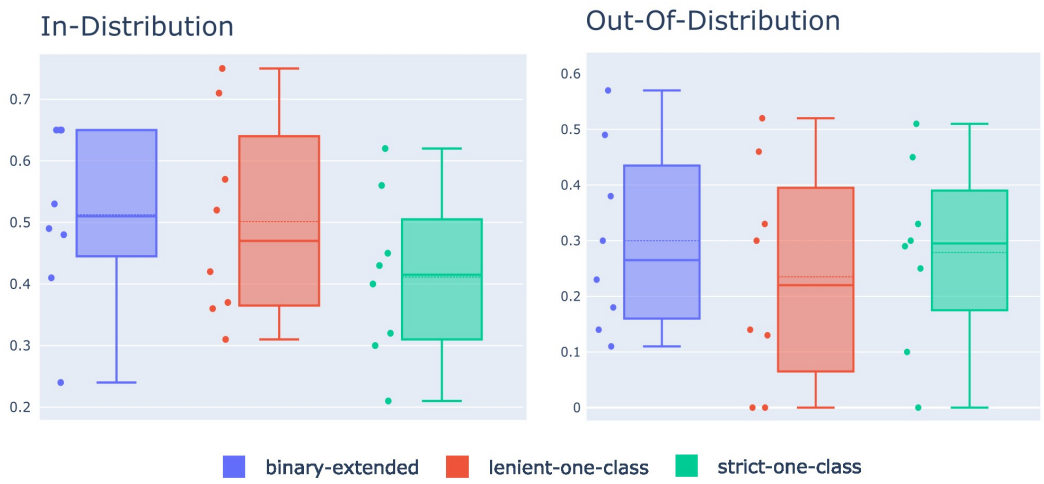

Figure 5.1: Mean F_1 scores across the different subcategories for the Out-Of-Distribution setting.

Because ambiguous boundaries can lead to misclassification or lower predictive accuracy, this limitation has an impact on the effectiveness of our models. Future research could address this limitation based on our observation that no single model is the best solution for the classification of all trigger warnings, but rather a combination of different approaches. While the other models could potentially balance this by being trained on either all of the annotated subcategories (Strict-One-Class) or at least one of the supercategories (Lenient-One-Class), which contain examples of subcategories that may also be positive for another subcategory of the same supercategory. As shown in Figures 5.2, model effectiveness varies greatly between subcategories and across setups within subcategories.
In the SVM experiments, for example, the mean F_1 for *Racism* was 0.07 for the Out-Of-Distribution setting and the Strict-One-Class setup, while the mean F_1 for *Violence* was 0.32 with the same setup and setting. The mean F_1 between subcategories for the fine-tuned RoBERTa model with the same setting and setup ranged from 0.10 to 0.51. This large variance indicates that there is no one solution for all subcategories. A combination of the best-performing models from each subcategory would undoubtedly improve the mean F_1 scores. Furthermore, we observed that models are able to generalise better within the *Aggression* supercategory. The generalisation for the Out-Of-Distribution setting is poor for *Discrimination* subcategories, particularly *Racism* and *Misogyny*, for both the SVM and the RoBERTa models. As a result of their keyword lists, these two subcategories contain a broader set of domains than the others.

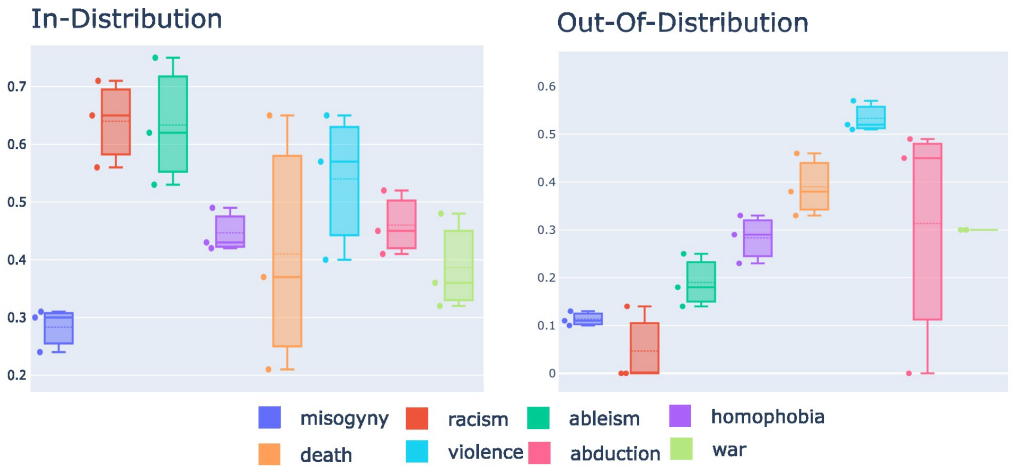

Figure 5.2: Mean F_1 scores across the different subcategories for the Out-Of-Distribution setting.

The keyword list from set 1 of the *Racism* subcategory consists primarily of racial slurs, whereas the list from set 2 consists primarily of tribal or historical words and phrases. In the In-Distribution setting the keyword lists of both sets are mixed up and, as a result, we see a significant increase in the effectiveness of the models for the *Racism* and *Misogyny* subcategories as well as the other *Discrimination* subcategories.

One significant limitation of our research, however, is that we did not take into account the intensity of potentially triggering events when annotating our retrieved examples. This omission represents a significant gap in our data collection process, as intensity can be a critical determinant in understanding how certain triggers affect individuals. The level of intensity can vary greatly, and this omission may result in an incomplete understanding of the triggers' effects. Future research should address this limitation by incorporating intensity assessment to provide a more thorough analysis of triggering events. Another limitation of our research is that the annotation process introduces our own biases into the dataset inadvertently. This problem is exacerbated by the previously mentioned lack of consideration for intensity, as the biases introduced during annotation may not adequately capture the nuance of the data. To address this limitation, stricter annotation guidelines and continuous quality checks and a more diverse group of annotators could be implemented to keep the impact of researcher bias on the dataset to a minimum.

Chapter 6

Conclusion

The primary goal of this study is to improve the process of creating training data so that a classifier can automatically identify and localise triggering concepts at the sentence level. This differs from the standard practice of addressing such concepts at the document level, aiming for greater precision and granularity. We constructed a dataset of 4,135 sentences in the English language, each manually labelled by three independent annotators and classified into one of eight distinct trigger warning concepts. A total of 12 experiments with two settings and three setups each were conducted using this dataset. The purpose of these experiments was to evaluate the performance of two distinct classification models, namely support vector machines (SVMs) and a fine-tuned, pre-trained BERT-based classifier (RoBERTa). Furthermore, the models' adaptability to new data domains was assessed. The research uncovered several key findings. The method of retrieving relevant sentences via keyword lists was found to be effective in identifying a large number of positive examples of triggering concepts. This highlights the importance of strategic keyword-based approaches for accurate concept identification within sentences. However, we also observed that there is no universally optimal model for trigger warning assignment. Rather, a combination of models tailored to specific warning categories emerged as a promising approach. This acknowledges the significant variations in model effectiveness across and within warning categories, given the large variations in effectiveness between and across warnings. In terms of model performance, the fine-tuned RoBERTa model outperformed the SVM, with an F_1 score of 0.30 in the In-Distribution setting and 0.51 in the Out-Of-Distribution setting. Furthermore, when compared to the SVM, this model demonstrated marginally better generalisation capabilities, with an improvement margin of 0.07.

Bibliography

Ying Chen, Sophia Yat Mei Lee, and Chu-Ren Huang. A cognitive-based annotation system for emotion computing. In *Proceedings of the Third Linguistic Annotation Workshop, LAW 2009, August 6-7, 2009, Singapore*, pages 1–9. The Association for Computer Linguistics, 2009. URL https://aclanthology.org/W09-3001/.

Alebachew Chiche and Betselot Yitagesu. Part of speech tagging: a systematic review of deep learning and machine learning approaches. *Journal of Big Data*, 9(1):1–25, 2022.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the international AAAI conference on web and social media*, volume 11, pages 512–515, 2017.

Mohammed Kaity and Vimala Balakrishnan. Sentiment lexicons and non-english languages: a survey. *Knowl. Inf. Syst.*, 62(12):4445–4480, 2020. URL https://doi.org/10.1007/s10115-020-01497-6.

Arunima Khunteta and Pardeep Singh. Emotion cause extraction - a review of various methods and corpora. In *2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)*, pages 314–319, 05 2021. doi: 10.1109/ICSCCC51823.2021.9478079.

Yoon Kim. Convolutional neural networks for sentence classification. *CoRR*, 2014. URL http://arxiv.org/abs/1408.5882.

Sophia Yat Mei Lee, Ying Chen, and Chu-Ren Huang. A text-driven rule-based system for emotion cause detection. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 45–53, Los Angeles, CA, 2010. Association for Computational Linguistics. URL https://aclanthology.org/W10-0206.

Leonard W. Levy, Kenneth L. Karst, and Dennis J. Mahoney, editors. *Encyclopedia of the American Constitution*. Macmillan, 2nd edition edition, 2000.

Bing Liu. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers, 2012. URL `https://doi.org/10.2200/S00416ED1V01Y201204HLT016`.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

Pushkar Mishra, Helen Yannakoudakis, and Ekaterina Shutova. Tackling online abuse: A survey of automated abuse detection methods. *CoRR*, 2019. URL `http://arxiv.org/abs/1908.06024`.

Alena Neviarouskaya and Masaki Aono. Extracting causes of emotions from text. In *Sixth International Joint Conference on Natural Language Processing, IJCNLP 2013, Nagoya, Japan, October 14-18, 2013*, pages 932–936. Asian Federation of Natural Language Processing / ACL, 2013. URL `https://aclanthology.org/I13-1121/`.

OpenAI. Chatgpt, 2023. URL `https://openai.com/research/chatgpt`. (Mar 14 version).

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.

Manuka Stratta, Julia Park, and Cooper deNicola. Automated content warnings for sensitive posts. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI EA '20, page 1–8, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450368193. doi: 10.1145/3334480.3383029. URL `https://doi.org/10.1145/3334480.3383029`.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. Label Studio: Data labeling software, 2020-2023. URL `https://github.com/heartexlabs/label-studio`. Open source software available from https://github.com/heartexlabs/label-studio.

Peter D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 417–424. ACL, 2002. doi: 10.3115/1073083.1073153. URL `https://aclanthology.org/P02-1053/`.

Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93, 2016.

Matti Wiegmann, Magdalena Wolska, Christopher Schröder, Ole Borchardt, Benno Stein, and Martin Potthast. Trigger warning assignment as a multi-label document classification problem. In *Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Toronto, Canada, July 2023. Association for Computational Linguistics.

Magdalena Wolska, Christopher Schröder, Ole Borchardt, Benno Stein, and Martin Potthast. Trigger warnings: Bootstrapping a violence detector for fanfiction. *arXiv preprint arXiv:2209.04409*, 2022.

Ali Yadollahi, Ameneh Gholipour Shahraki, and Osmar R. Zaïane. Current state of text sentiment analysis from opinion to emotion mining. *ACM Comput. Surv.*, 50(2):25:1–25:33, 2017. doi: 10.1145/3057270. URL `https://doi.org/10.1145/3057270`.