The Pearl Retriever: Two-Stage Retrieval for Pairs of Argumentative Sentences

Sebastian Schmidt, Jonas Probst, Bianca Bartelt, and Alexander Hinz

08.09.2022

Retrieval is split into separate pipelines for arguments and sentences

Arguments

Index

- 60,000 non-argumentative „arguments“ were discarded based on quality scores
  - Classification based on the Webis Argument Quality Corpus 2020 (Gienapp, L., et al. 2020)
- Index on premises and the conclusion

- DirichletLM (Zhai, C., Lafferty, J., 2017)
- Highly relevant results with little variance (Potthast, M. et al. 2019)

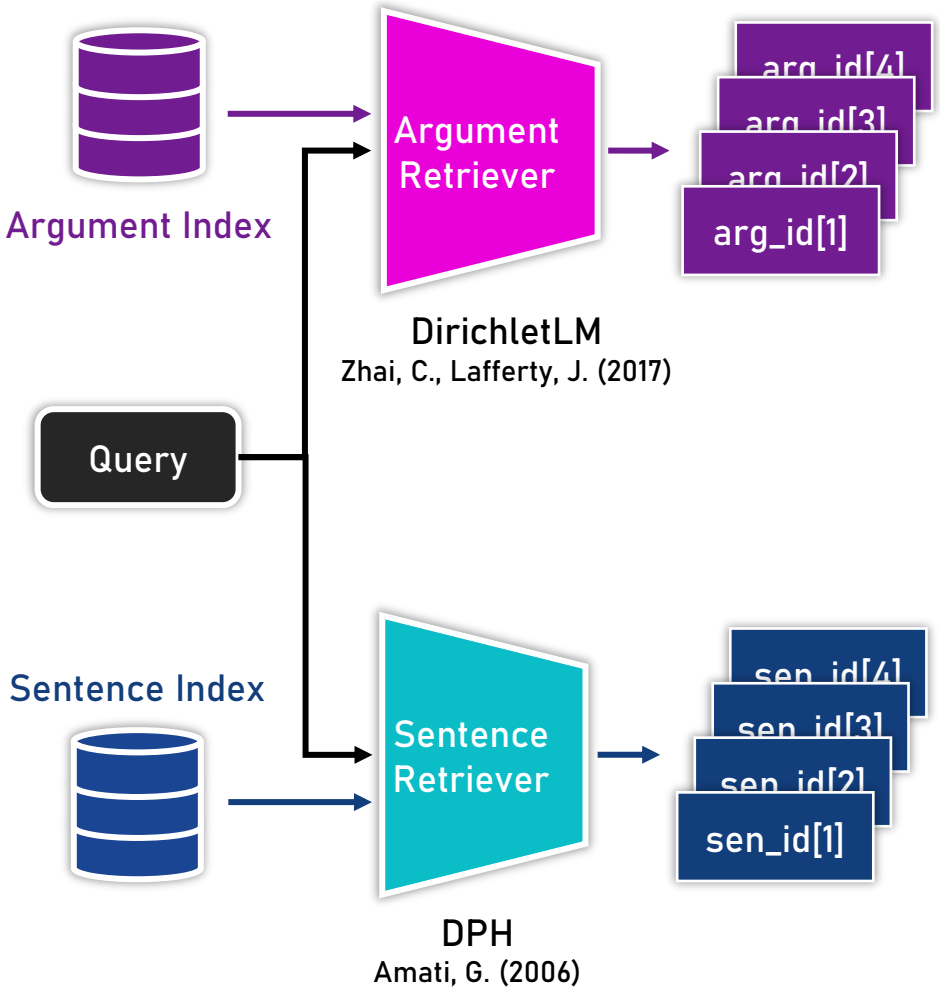
Sentences

Index

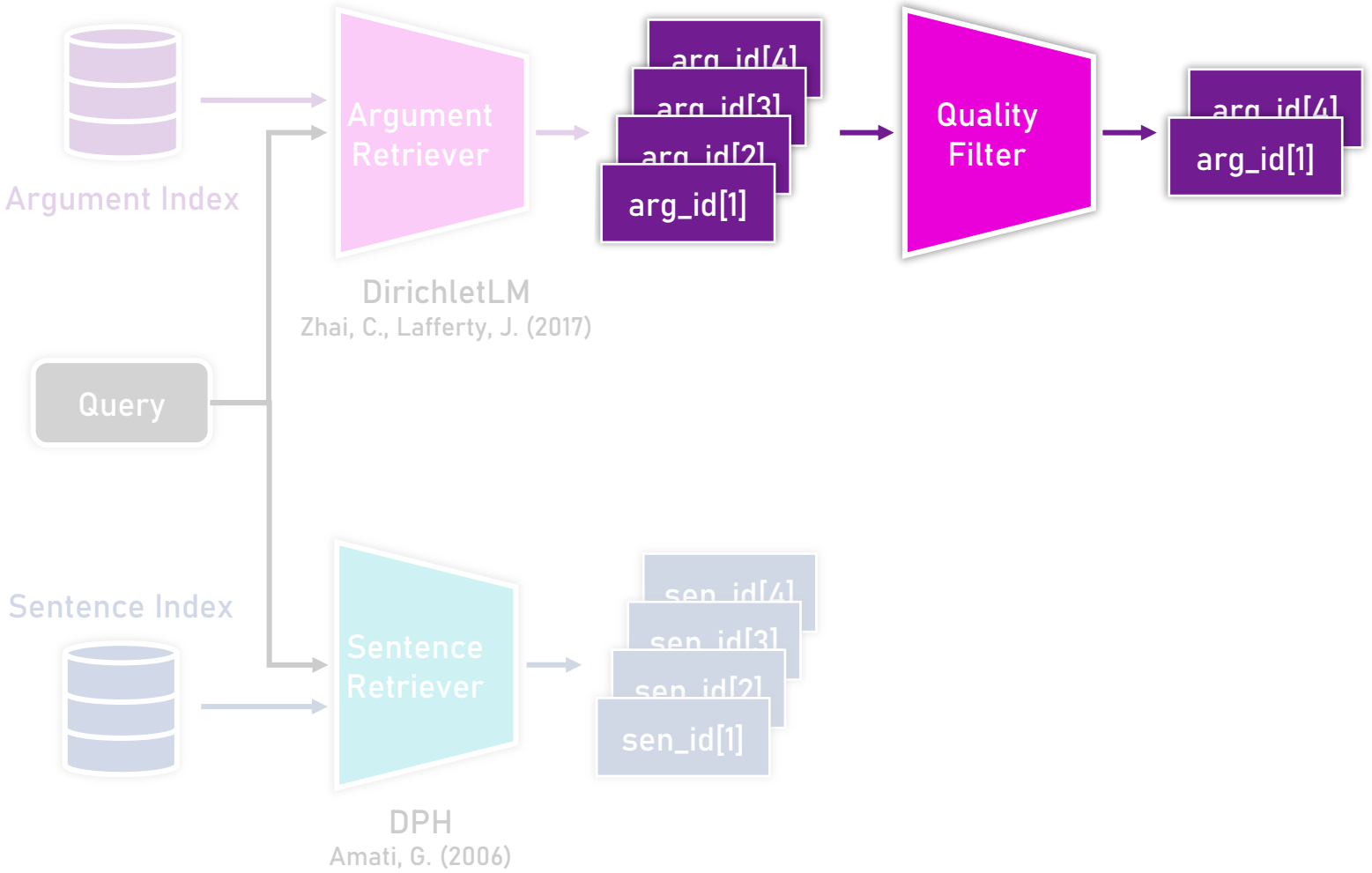
- Individual sentences
- Index on sentence content

- DPH-Retrieval (Amati, G., 2006)
- Highly relevant results (Potthast, M. et al. 2019)
- Focus on specific query terms

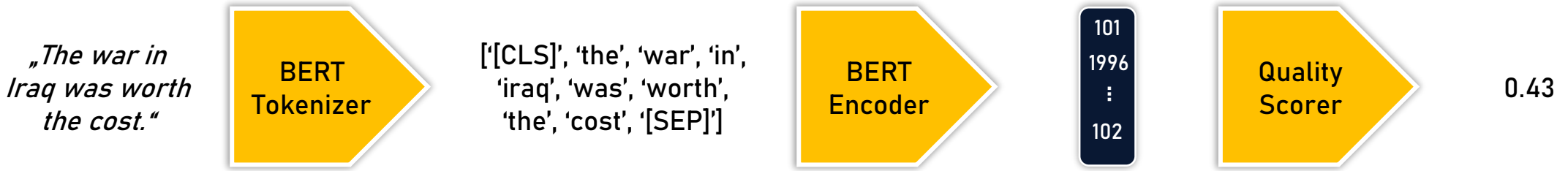
First, a selection of relevant arguments and sentences is retrieved



Then, arguments are filtered according to their quality



The quality scores are calculated on BERT encodings



Arguments

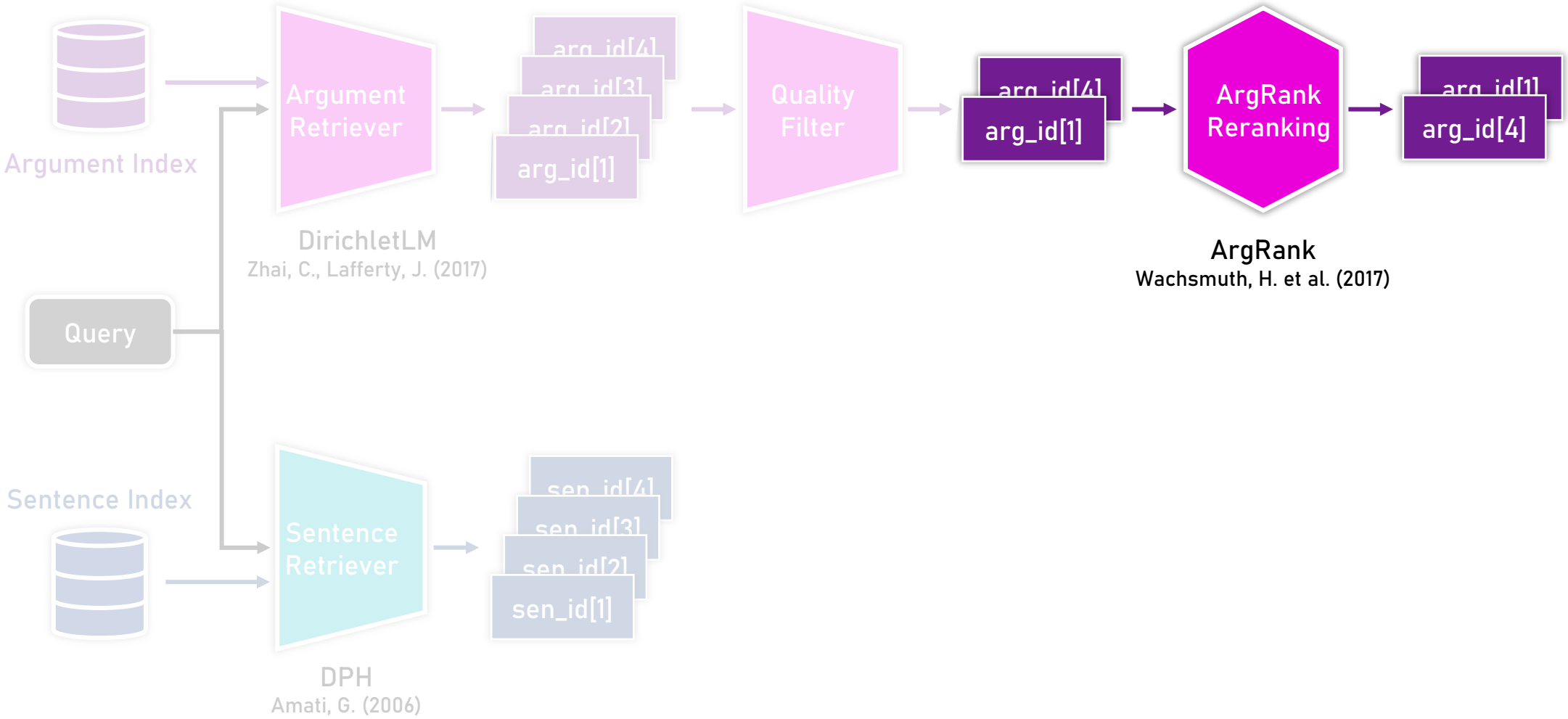
- Webis Argument Quality Corpus 2020 (Gienapp, L., et al. 2020)
- Scores between -4 and 3
- Arguments with a score < 1 are discarded
- Validation performance was consistent with Team Yeagerists in Touché 2021 (Bondarenko, A. et al. 2021)

Sentences

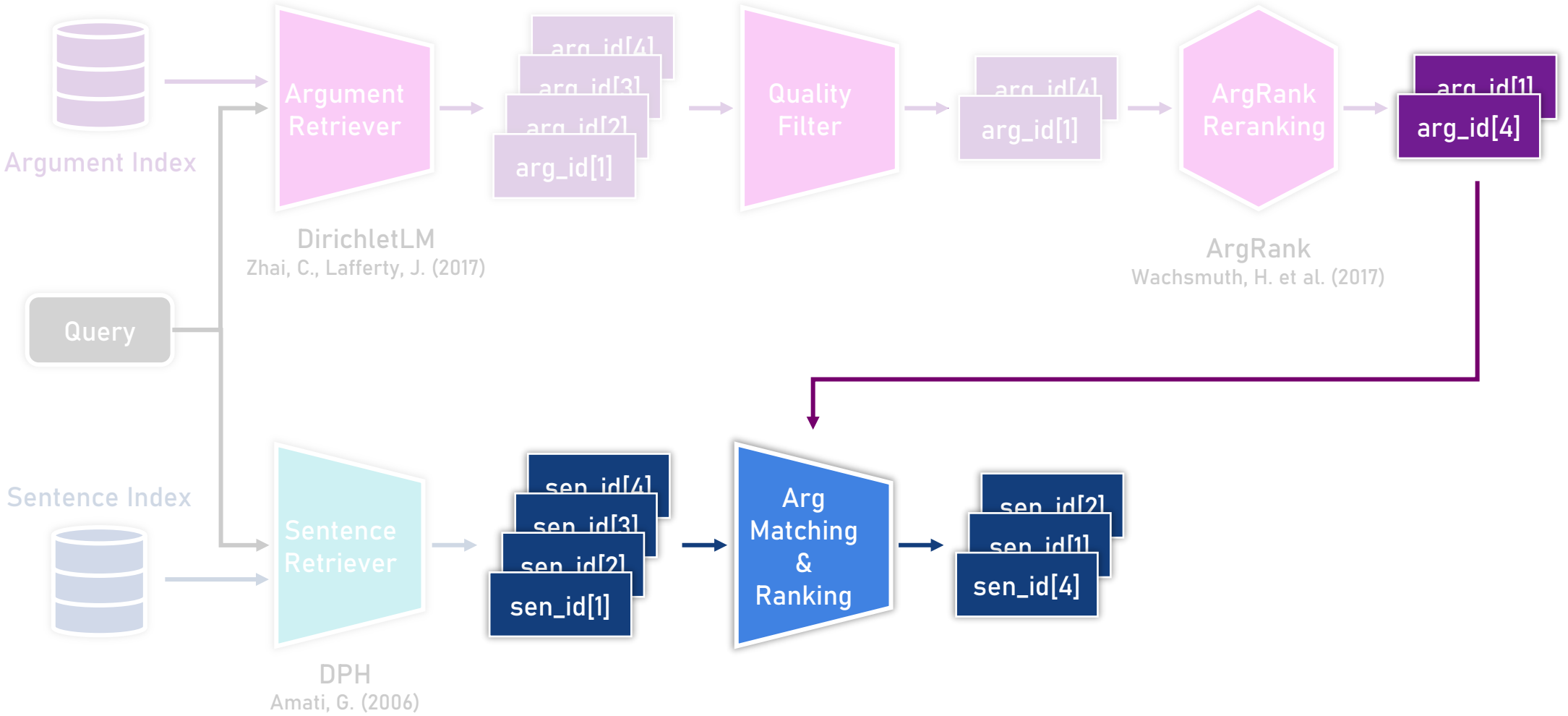
- IBM Debater - IBM-ArgQ-Rank-30kArgs (Gretz, S. et al. 2019)
- Scores between 0 and 1
- Sentences with a score < 0.7 are discarded
- Validation-Performance consistent with baseline by Gretz, S. et al. (2019)

- Architecture based on Gretz, S. et al. (2019)
- Implemented with PyTorch and the Hugging Face Library

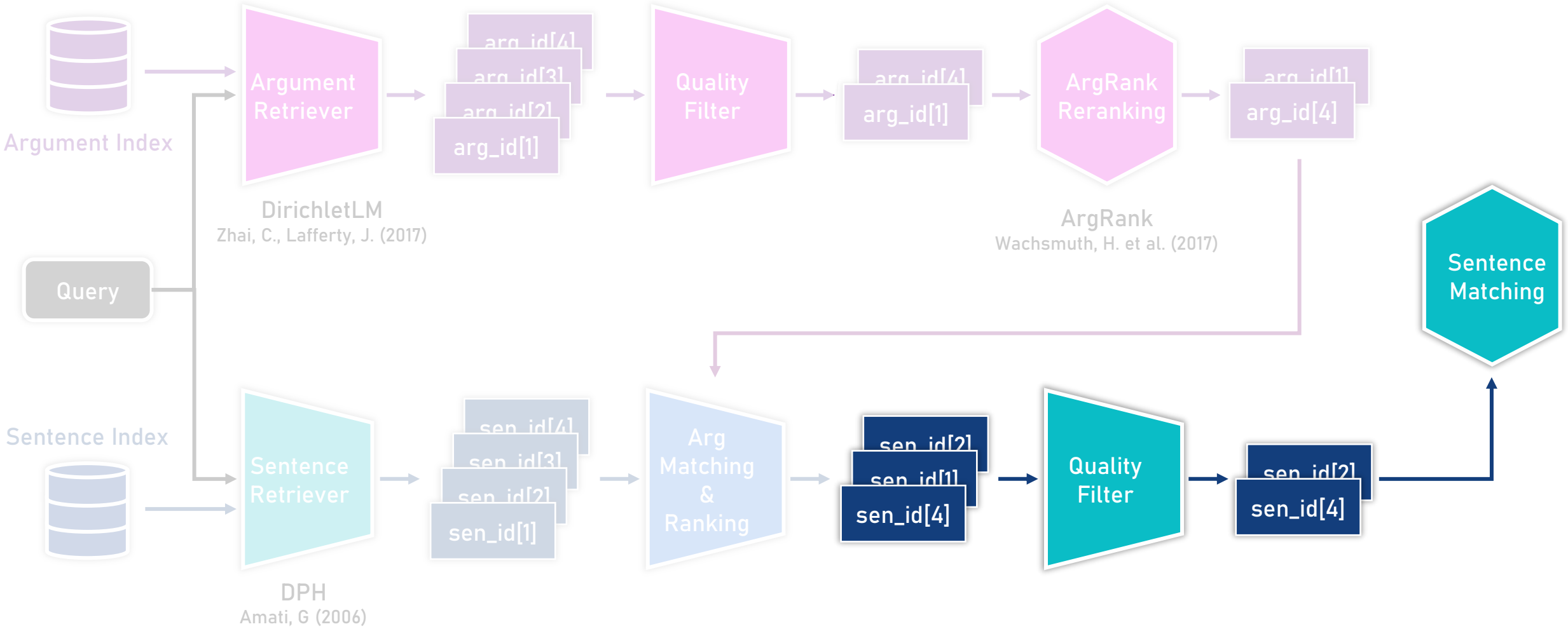
ArgRanks are used to rerank the remaining arguments



Sentences are filtered and sorted based on their source arguments

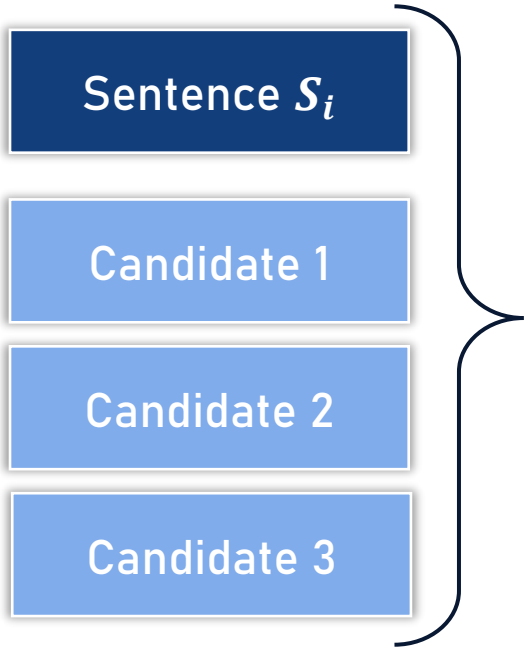


After filtering on quality, the final sentence pairs are formed



Two approaches to sentence matching were evaluated (I/II)

The first approach was inspired by Maximal Marginal Relevance (MMR)[1]



Retrieval Result R

Choose partner S_j for sentence S_i, that fulfills:

$$\max_{S_j \in R \setminus \{S_i\}} \left[\lambda * sim_1(S_i, S_j) - (1-\lambda) * sim_2(S_i, S_j)\right]$$

$sim_1(S_i, S_j)$ Next Sentence Prediction (L2-Normalized)

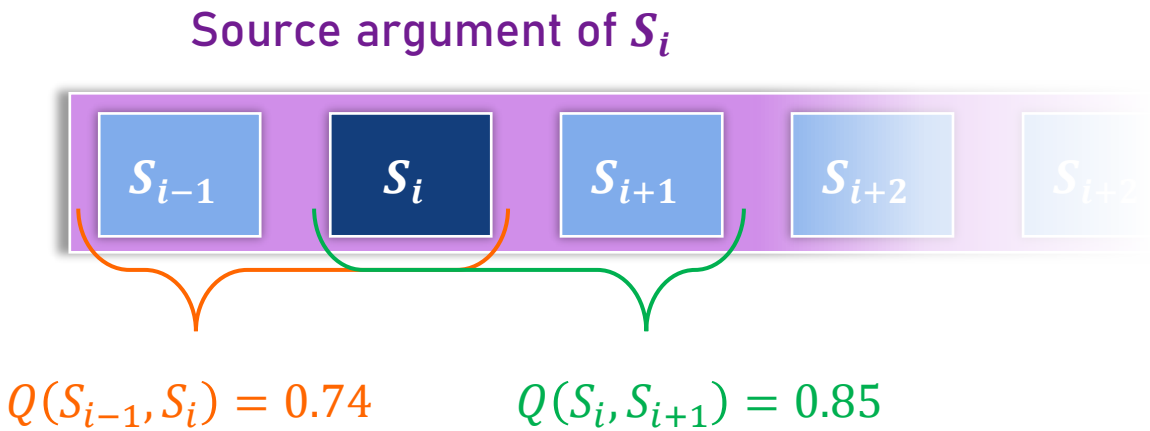
$sim_2(S_i, S_j)$ Cosine similarity of S_i and S_j

- No sentence is matched with multiple other sentences
- λ=0.5 leads to nDCG = 0.2801
- λ=1 (Next Sentence Prediction) leads to nDCG = 0.4255

1: Carbonell, J. & Goldstein, J. (1998)

Two approaches to sentence matching were evaluated (II/II)

The second approach forms pairs within existing arguments



Quality scores are calculated for pairs with the preceding and following sentence

Neighbor Matching

- Quality scores are calculated using the sentence quality model
- The optimal neighbor was precalculated for each sentence
- nDCG = 0.6593

Extension with a blocklist

- Sentences that include passages like “My opponent” are discarded
- Improvement to nDCG = 0.6914

Our evaluation was based on three metrics

Every retrieval result was evaluated on

- Argumentativeness $r_a \in \{-2, 0, 1, 2, 3\}$
- Sentence Coherence $r_c \in \{-2, 0, 1, 2, 3\}$
- Argument Representation $r_r \in \{-2, 0, 1, 2, 3\}$

The final nDCG for each model is calculated as the average nDCG over 10 queries and three metrics:

$$nDCG = \frac{nDCG_a + nDCG_c + nDCG_r}{3}$$

The optimal ranking was chosen for each query and metric individually

The final nDCG is based on the average of the three metrics

	Prototype	Neighbor Matching	Blocklist	ArgRank
Argumentativeness	0.3997	0.5814	0.6281	0.6168
Sentence Coherence	0.3966	0.7782	0.7814	0.7792
Argument Representation	0.6967	0.6184	0.6648	0.6873
nDCG@10	0.4977	0.6593	0.6914	0.6944
Variance	0.029710	0.010935	0.006412	0.006628

Each model includes the methods of the previous level

We draw three main takeaways from our experiments

Simple solutions were able to achieve good results

- Neighbor-Matching nDCG = 0.6593 vs. Next Sentence Prediction nDCG = 0.4255
- The blocklist led to a noticeable improvement on nDCG

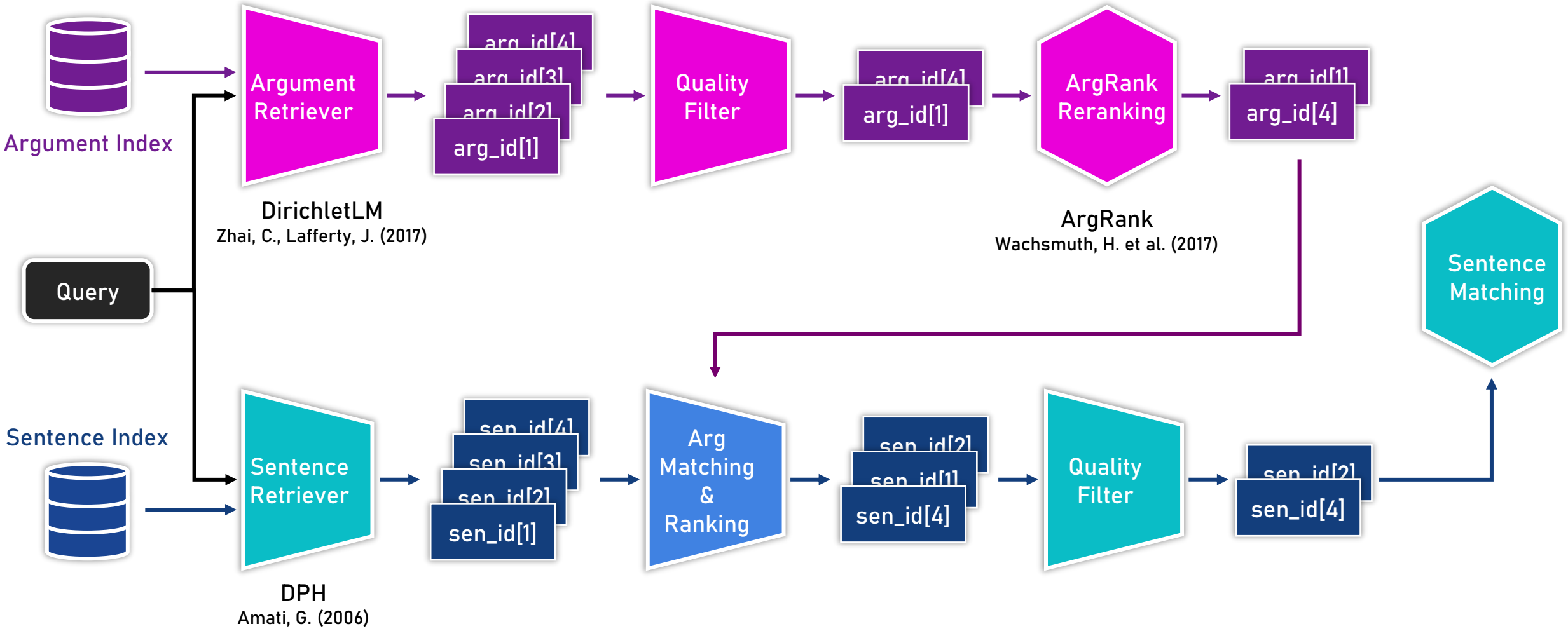
ArgRank had only very little influence in our experiments

- Possible explanation: Low edge density (44,250 edges for roughly 300,000 arguments)

DPH's strong focus on specific query terms can be disadvantageous

- „9/11 was an inside job" is retrieved for the query „Should Insider Trading Be Allowed?"
- This influence is reduced by the more stable argument retrieval using Dirichlet

The full retrieval architecture



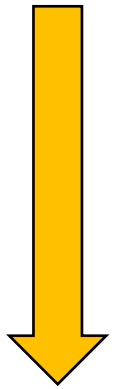
Backup

Results achieved on the official evaluation

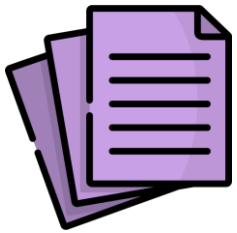
	Blocklist	ArgRank
Quality	0.670	0.678
Sentence Coherence	0.392	0.398
Relevance	0.481	0.479
nDCG@10	0,5143	0.5183

The argument graph (ArgGraph) forms the basis for ArgRank

Conclusion: "Grey imports limit a company's control over its own products"
Discussion: "Allow retailers to import for resale 'grey' goods from abroad."
Stance: CON



1. Identfication of premises from arguments with the same stance in the same discussion
2. Calculation of encodings for conclusion and premises with MPNET [Song, K. et al. (2020)]
3. Creation of edges for arguments with cosine similarity > 0.7

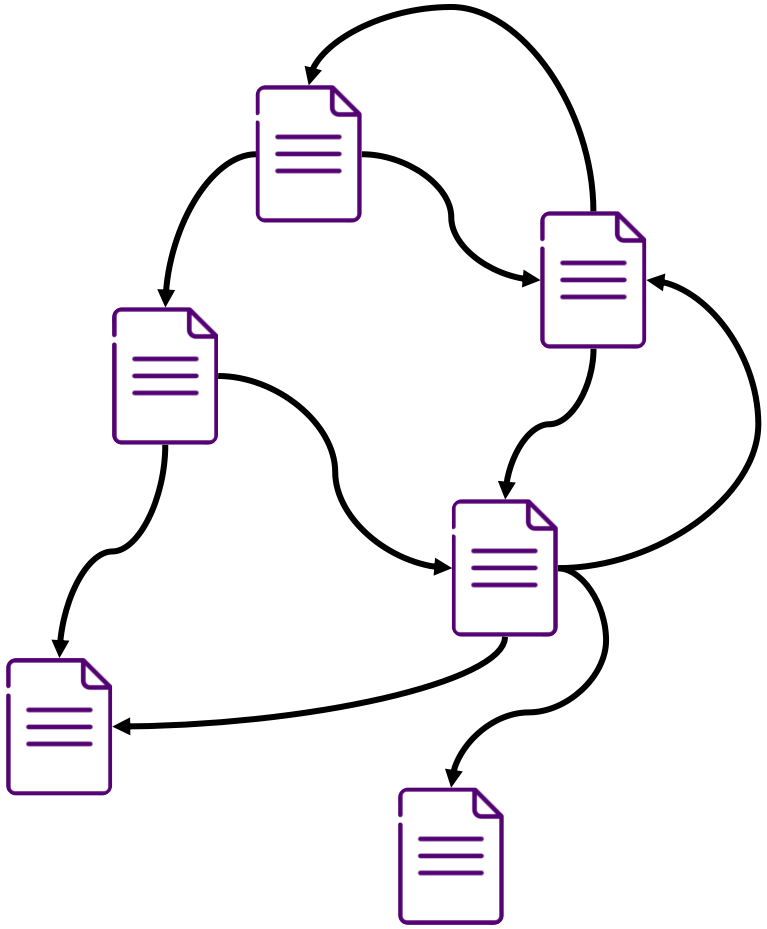


Premise: "Grey imports result in the manufacturer/ distributor effectively losing some, and often most, control of their pricing and retailing strategy in the importing country."
Similarity = 0.80131
Premises = 5

Premise: "The loss of revenue from grey imports can mean that production is limited or even halted going forward, even though there is market demand for more products from the manufacturer."
Similarity = 0.79121
Premises = 4

ArgRank Source: Wachsmuth, H. et al. (2017)

The ArgRank was calculated based on the identified graph



ArgGraph for cosine similarity > 0.7

Interpretation: Arrow from A_1 to A_2:
Argument A_1 uses the conclusion of A_2 as premise

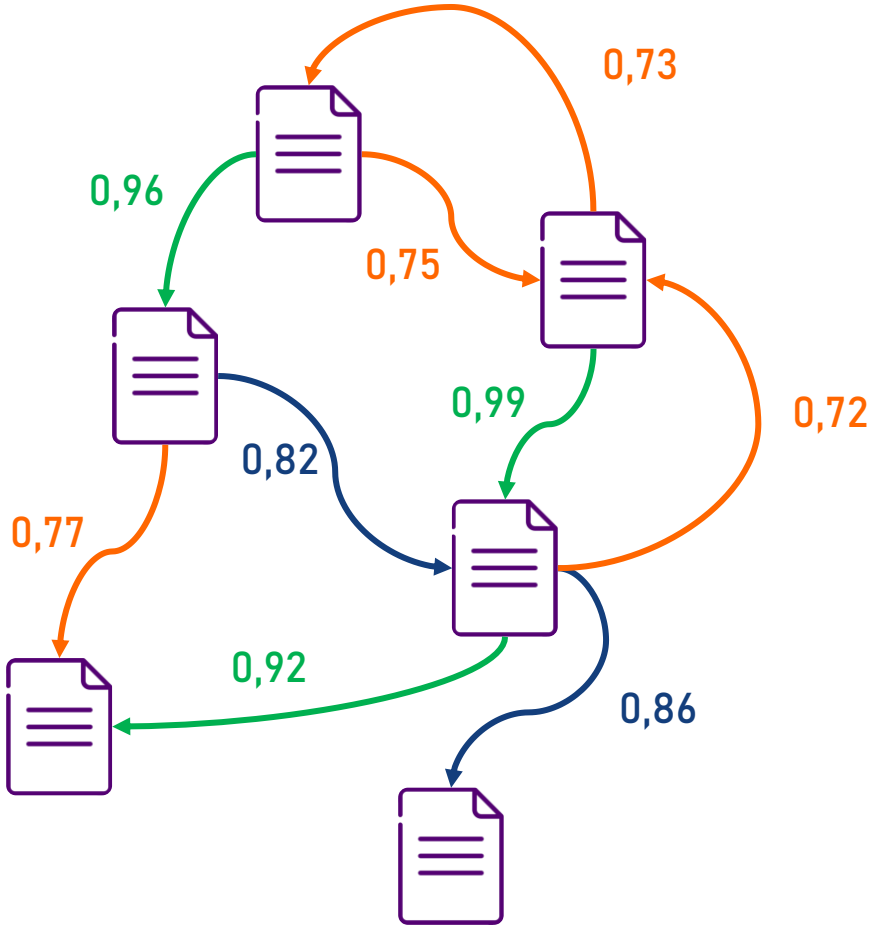
Adjusted calculation of the ArgRank for argument i:

$$p(c_i) = \frac{(1-\alpha)}{|A|} + \alpha * \sum_j \frac{p(c_j)}{|P_j|} * sim(c_i, p_{jk})$$

$p(c_i)$	ArgRank of argument
$\|A\|$	Number of arguments in the corpus
$p(c_j)$	ArgRank of argument j, that uses the conclusion c_i
$\|P_j\|$	Number of premises of argument j
$sim(c_i, p_{jk})$	Similarity between c_i and the premise [Optional]

ArgRank Source: Wachsmuth, H. et al. (2017)

The ArgRank was calculated based on the identified graph



ArgGraph for cosine similarity > 0.7

Interpretation: Arrow from A_1 to A_2:
Argument A_1 uses the conclusion of A_2 as premise

Adjusted calculation of the ArgRank for argument i:

$$p(c_i) = \frac{(1-\alpha)}{|A|} + \alpha * \sum_j \frac{p(c_j)}{|P_j|} * sim(c_i, p_{jk})$$

Several versions of the ArgGraph were created based on cosine similarity:

- Similarity > 0.9
- Similarity > 0.8
- Similarity > 0.7

The highest nDCGs were achieved for

- $\alpha = 0{,}3$ and similarity > 0.75 (nDCG = 0.6944)
- $\alpha = 0{,}4$ and similarity > 0.8 (nDCG = 0.6924)

ArgRank Source: Wachsmuth, H. et al. (2017)

Sources

Sources (1/3)

- Amati, G. (2003):
  - Probability models for information retrieval based on divergence from randomness (Doctoral dissertation, University of Glasgow).

- Amati, G. (2006):
  - Frequentist and Bayesian Approach to Information Retrieval. Proceedings of the 28th European conference on Advances in Information Retrieval pp. 13-24. DOI: 10.1007/11735106_3.

- Gretz, S., Friedman, R., Cohen-Karlik, E., Toledo, A., Lahav, D., Aharonov, R., Slonim, N. (2019):
  - A Large-scale Dataset for Argument Quality Ranking: Construction and Analysis. arXiv:1911.11408.

- Bondarenko, A., Gienapp, L., Fröbe, M., Beloucif, M., Ajjour, Y., Panchenko, A., Biemann, C., Stein, B., Wachsmuth, H., Potthast, M., Hagen, M. (2021):
  - Overview of Touché 2021: Argument Retrieval. Experimental IR Meets Multilinguality, Multimodality, and Interaction. DOI: 10.1007/978-3-030-85251-1_28.

Sources (2/3)

- Carbonell, J. & Goldstein, J. (2017):
  - The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. SIGIR Forum 51, 2, 209–210. DOI: https://doi.org/10.1145/3130348.3130369.

- Gienapp, L., Stein, B., Hagen, M., Potthast, M. (2020):
  - Webis Argument Quality Corpus 2020 (Webis-ArgQuality-20) (1.0.0) [Data set]. 2020 Annual Conference of the Association for Computational Linguistics (ACL 2020), Seattle. Zenodo. https://doi.org/10.5281/zenodo.3780049

- Potthast, M., Gienapp, L., Euchner, F., Heilenkötter, N., Weidmann, N., Wachsmuth, H., Stein, B., Hagen, M. (2019):
  - Argument Search: Assessing Argument Relevance. The 42nd International ACM SIGIR Conference 1117-1120. DOI: 10.1145/3331184.3331327.

- Song, K., Xu, T., Qin, T., Lu, J., Liu, T.-Y. (2020):
  - MPNet: Masked and Permuted Pre-training for Language Understanding. arXiv:2004.09297.

Sources (3/3)

- Wachsmuth, H., Stein, B., Ajjour, Y. (2017):
  - “PageRank” for Argument Relevance. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pp. 1117-1127.
- Zhai, C., Lafferty, J. (2017):
  - A Study of Smooting Methods for Language Models to Ad Hoc Information Retrieval. SIGIR Forum 51, 2 (July 2017), pp. 268–276. DOI: 10.1145/3130348.3130377.