Overview of Touché 2020: Argument Retrieval

Alexander Bondarenko,[1] Maik Fröbe,[1] Meriem Beloucif,[2] Lukas Gienapp,[3]
Yamen Ajjour,[1] Alexander Panchenko,[4] Chris Biemann,[2] Benno Stein,[5]
Henning Wachsmuth,[6] Martin Potthast,[3] and Matthias Hagen[1]

[1]Martin-Luther-Universität Halle-Wittenberg
[2]Universität Hamburg
[3]Leipzig University
[4]Skolkovo Institute of Science and Technology
[5]Bauhaus-Universität Weimar
[6]Paderborn University

touche@webis.de touche.webis.de

Abstract Argumentation is essential for opinion formation when it comes to debating on socially important topics as well as when making everyday personal decisions. The web provides an enormous source of argumentative texts, where well-reasoned argumentations are mixed with biased, faked, and populist ones. The research direction of developing argument retrieval technologies thus focuses not only retrieving relevant arguments for some argumentative information need, but also on retrieving arguments of a high quality.

In this overview of the first shared task on argument retrieval at the CLEF 2020 Touché lab, we survey and evaluate 41 approaches submitted by 17 participating teams for two tasks: (1) retrieval of arguments on socially important topics, and (2) retrieval of arguments on everyday personal decisions.

The most effective approaches submitted share some common techniques, such as query expansion, and taking argument quality into account. Still, the evaluation results show that only few of the submitted approaches (slightly) improve upon relatively simple argumentation-agnostic baselines—indicating that argument retrieval is in its infancy and meriting further research into this direction.

1 Introduction

Decision making and opinion formation processes are routine tasks for most of us. Often, such opinion formation relates to a decision between two sides based on previous experience and knowledge, but it may also require accumulating new knowledge. With the widespread access to every kind of information on the web, everyone has the chance to acquire new knowledge and to form an informed opinion about a given topic. In the process, be it on the level of socially important topics or "just" personal decisions, one of the at least two sides (i.e., decision options) will challenge the other with an appeal to justify its stance. In the simplest form, a justification might be simple facts or opinions, but more complex justifications often are based on argumentation: a complex relational aggregation of evidence and opinions, where one element is supported by the other.

Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0). CLEF 2020, 22-25 September, Thessaloniki, Greece.

CEUR-WS.org/Vol-2696/paper_261.pdf

Web resources, such as blogs, community question answering sites, and social platforms, contain an immense variety of opinions and argumentative texts—including many that can be considered to be of a biased, faked, or populist nature. This motivates research on the task of argument retrieval, which concerns specific informations needs that current web search engines, as far as we can tell, do not treat any differently than "standard" ad hoc search.[1,2] One of the first argument search engines has been args.me [45], which retrieves relevant arguments on a given (controversial) query from a focused collection of arguments crawled from a selection of debate portals [45]. Other argument retrieval systems, such as ArgumenText [40] and TARGER [9], use the larger Common Crawl, requiring additionally also argument mining components as part of their retrieval pipelines. The comparative argumentation machine CAM [37] also aims at supporting decision making in comparison scenarios, based on billions of sentences from the Common Crawl, however, it still lacks a proper ranking of diverse arguments. Moreover, argument retrieval is not just limited to search engines serving more argumentative results for corresponding information needs: it will also be an integral part of any open-domain conversational agent capable of "discussing" controversial topics with their human users—as showcased by IBM's Project Debater [5, 23].[3]

To foster research on argument retrieval, we organize the Touché lab at CLEF 2020 and the first shared task on argument retrieval.[4] The lab is a collaborative platform to develop argument retrieval approaches for decision support on a societal (e.g.,"Is climate change real and what to do?") and also on a personal level (e.g.,"Should I buy real estate or rent, and why?"), featuring two tasks:

1. Given a focused collection of arguments and some socially important and controversial topic, retrieve arguments that could help an individual forming an opinion on the topic, or arguments that support/challenge their existing stance.
2. Given a generic web crawl and a comparative question relating to a personal decision, retrieve documents with arguments that could help an individual to arrive at a conclusion regarding their decision.

From the 28 teams registered for the Touché lab (10 teams for Task 1, 7 for Task 2, and 11 for both) 17 teams actively participated with a total of 41 different approaches (to allow for a wide diversity of ideas, the teams could submit multiple approaches). Additionally, we evaluated two retrieval baselines: the Lucene implementation of query likelihood with Dirichlet-smoothed language models (DirichletLM [48]) for Task 1 and the BM25F-based [36] search engine ChatNoir [6] for Task 2. Effectiveness was measured using nDCG@5 [17] based on the top-5 pools manually labeled for relevance by human assessors (7,045 judgments in total). The most effective approaches in both tasks use query expansion and take argument quality into account, but the evaluation results show that only few of the submitted approaches (slightly) improve upon relatively simple baselines. Further research on argument retrieval thus seems well-justified.

[1] A notable exception have been Bing's "Multi-Perspective Answers".

[2] https://blogs.bing.com/search-quality-insights/february-2018/Toward-a-More-Intelligent-Search-Bing-Multi-Perspective-Answers

[3] https://www.research.ibm.com/artificial-intelligence/project-debater/

[4] 'touché' is commonly "used to acknowledge a hit in fencing or the success or appropriateness of an argument, an accusation, or a witty point." [https://merriam-webster.com/dictionary/touche]

2 Related Work

Examples of argument retrieval scenarios are opinion formation on controversial topics [40, 45] or on comparison options [37] but also finding counterarguments for a given argument [47]. In the Touché lab, we address the first two types of information needs in two separate tasks. Here, we briefly review the related work on argument retrieval in general and on retrieval in comparative scenarios.

2.1 Argument Retrieval

Usually, an argument is modeled as a conclusion with supporting or attacking premises [45]. While a conclusion is a statement that can be accepted or rejected, a premise is a more grounded statement (e.g., a statistical evidence). The development of an argument retrieval system comes with challenges that range from mining arguments from unstructured text to assessing their relevance and quality [45].

Argument retrieval can follow different paradigms that start from different sources and employ argument mining and retrieval tasks in different orders [2]. For instance, the args.me search engine's approach [45] is based on a focused crawl of arguments from online debate portals that were gathered in an offline pre-processing using tailored heuristics. In the online query phase, the extracted arguments are then ranked using BM25F [36] giving conclusions more weight than premises. Also Levy et al. [21] use distant-supervision to mine arguments in an offline pre-processing for a set of topics from Wikipedia before ranking them. In contrast, the ArgumenText search engine [40] and the TARGER argument retrieval component [9] are based on Common Crawl[5] web crawls without an offline pre-processing phase for argument mining. Both systems simply retrieve web documents and then use argument mining approaches in an online manner to extract arguments from the retrieved documents. The two tasks in the Touché lab address the two different paradigms of a focused argument crawl, similar to that of the args.me search engine, as the to-be-indexed dataset (Task 1), and of a general web crawl from which argumentative results are to be retrieved (Task 2).

Apart from an argument's topical relevance, argument retrieval might also need to take an argument's quality into account. What makes a good argument has been studied since the time of Aristotle [4]. Recently, Wachsmuth et al. [43] categorized the different aspects of argument quality into a taxonomy that covers three quality dimensions: logic, rhetoric, and dialectic quality. Logic quality refers to the local structure of an argument, i.e, the conclusion and the premises and their relations. Rhetoric quality covers the effectiveness of an argument in persuading an audience with its conclusion. Dialectic quality addresses the relations of an argument to other arguments on the topic. For example, an argument may be particularly vulnerable in a debate when many attacking arguments exist. Note that the topical relevance of an argument is part of the dialectic quality in Wachsmuth et al.'s categorization [43].

In an evaluation of standard text-based retrieval models on an argument collection, Potthast et al. [31] incorporate argument quality evaluation with respect to the aforementioned dimensions. Based on a TREC-style pooling and human assessment of relevance as well as the three dimensions, DirichletLM turns out to be better suited for

[5] http://commoncrawl.org

argument retrieval than BM25, DPH, and TF-IDF. Later, Gienapp et al. [14] suggested a strategy to annotate the argument quality dimensions in a pairwise manner, reducing annotation costs by 93%.

Apart from standard text-based retrieval models, also other ideas have been proposed for argument ranking. For example, Wachsmuth et al. [46] suggest to create argument graphs by connecting two arguments if one uses the other's conclusion as a premise, and to exploit this structure for argument ranking using PageRank [27]. Another example is Dumani et al.'s idea of a probabilistic framework that clusters semantically similar claims and premises [11], taking into account potential support/attack relations between premise and claim clusters, and claim clusters and a query.

2.2 Retrieval in Comparative Scenarios

Comparative information needs arise when someone has to decide between different options. Such needs range from simple facts (e.g., "Does water or ethanol have the higher boiling point?") over comparing products (e.g., "What is the best phone for me?") to problems like which location or college to choose for undergraduate studies.

The earliest web-based comparison systems employed interfaces where users entered the to-be-compared objects into dedicated search boxes [25, 42]. In parallel to the early comparison system development, opinion mining from product reviews has been dealing with the identification of comparative sentences and their polarity (in favor or not) using various techniques [18, 19, 20]. Recently, the identification of preferences ("winning" object) from comparative sentences has been addressed in open-domain settings (not just for product reviews) by applying feature-based and neural classifiers [22, 29]. This preference classification forms the basis of the comparative argumentation machine CAM [37], which accepts two to-be-compared objects and comparison aspects as input, retrieves comparative sentences in favor of one or the other object using BM25, and clusters the sentences to present a summary table. Improving retrieval models for systems like CAM is the objective of Touché's Task 2.

3 Lab Overview and Statistics

A total of 28 teams registered for the Touché lab, with a majority coming from Germany (17 teams from Germany, two from France, two from India, and one each from China, Italy, the Netherlands, Pakistan, Russia, Switzerland, and the US). For a nice team naming scheme, participants could choose as their team name a real or fictional fencer or swordsman (e.g., Zorro)—a tip of the hat to 'touché's other meaning.

From the 28 registered teams, 17 actively participated in the lab by submitting approaches/results. We asked the teams to use the TIRA evaluation platform [32], enabling the submission of working software, in order to increase the overall result reproducibility. TIRA is an integrated cloud-based evaluation-as-a-service shared task platform where teams have full administrative access to a virtual machine. By default, the virtual machines operated Ubuntu 18.04 with one CPU core (Intel Xeon E5-2620), 4GB of RAM, and 16GB HDD, but we adjusted the resources to the participants' requirements when needed (e.g., one team asked for 24 GB of RAM, 5 CPU cores, and

30 GB of HDD). Each virtual machine had standard software pre-installed (e.g., Docker and Python) to simplify the deployment. After deployment, the teams could create result submissions via the web UI of TIRA, triggering the following standard pipeline. To create a run submission from a participating team's software, the respective virtual machine was shut down, disconnected from the internet, cloned, and the clone booted up again, this time mounting also the test datasets for the respective task. The interruption of the internet connection was meant to discourage the use of external web services, which may disappear or get incompatible in the future, harming reproducibility. However, two exceptions were made for all participants: the APIs of ChatNoir and args.me were available, even in the sandbox mode. Additionally, if participants requested it, other external web services based on the teams' requirements could be whitelisted; only one team asked to access the Web of Trust API.[6] All virtual machines that the teams used for their submissions are archived such that they can be re-evaluated or applied to new datasets as long as data formats and external APIs remain stable.

To foster diverse approaches, the participating teams could submit several runs, giving priorities for evaluation when more than one was submitted. The actual output run files needed to follow the standard TREC-style format.[7] Upon submission, we checked the validity of the run files and asked the participants to adjust and re-submit their runs in case of problems, also offering assistance. This resulted in 41 valid runs from 17 teams. From every team, at least the top five runs of highest priority were pooled for evaluation. Additionally, we evaluated two retrieval baselines: the Lucene implementation of query likelihood with Dirichlet-smoothed language models (DirichletLM [48]) for Task 1, and the BM25F-based [36] search engine ChatNoir [6] for Task 2.

4 Touché Task 1: Conversational Argument Retrieval

The goal of the Touché's first task is to provide assistance to users searching for good and relevant pro and con arguments on various societal topics (e.g., climate change, electric cars, etc.) that are, for instance, engaged in an argumentative conversation. A respective retrieval system may aid users in collecting evidence on issues of general societal interest and support them in forming their own opinion.

Several existing community question answering websites, such as Yahoo! Answers and Quora, as well as debate portals, such as debatewise.org and idebate.org, are designed to accumulate opinions and arguments and to engage users in dialogues. Generic web search engines lack an effective solution to retrieve relevant arguments from these and other platforms beyond things like returning entire discussion threads. One reason lies in their ignorance of the argumentative nature of the underlying discussions, which results in generic web search engines offering only limited support during conversations or debates. This motivates the development of robust and effective approaches specifically focused on conversational argument retrieval.

[6] https://www.mywot.com/developers

[7] Also described on the lab website: https://touche.webis.de

Table 1. Example topic for Task 1: Conversational Argument Retrieval

Number	21
Title	Is human activity primarily responsible for global climate change?
Description	As the evidence that the climate is changing rapidly mounts, a user questions the common belief that climate change is anthropogenic and desires to know whether humans are the primary cause, or whether there are other causes.
Narrative	Highly relevant arguments include those that take a stance in favor of or opposed to climate change being anthropogenic and that offer valid reasons for either stance. Relevant arguments talk about human or non-human causes, but not about primary causes. Irrelevant arguments include ones that deny climate change.

4.1 Task Definition

The participants of Task 1 are asked to retrieve relevant arguments from a focused crawl of arguments originating from debate portals for a given query on some controversial topic. Given the amount of argumentative texts readily available on such portals, instead of having to develop own argument extraction technology, the participants could focus their attention on retrieving the previously extracted arguments from the portals, covering a wide range of popular debate topics. To ease access to the argument collection, we provide an openly accessible and flexible API at args.me,[8] also allowing participants to participate in the lab without having to index the collection themselves.

4.2 Data Description

Retrieval Topics. We have formulated 50 search scenarios on controversial issues in the form of TREC-style topics with a title (the query potentially issued by a user), a description (a short summary of the search context and information need), and a narrative (a definition of what constitutes relevant results for this topic, serving as a guideline for human assessors). An example topic is shown in Table 1. As topics, we selected those issues that have the largest number of user-generated arguments on the debate portals, and thus can be assumed to be of high societal interest. Further, we ensured that, for each topic, at least some relevant arguments are present in the collection.

Document Collection. Task 1 is based on the args.me corpus [2], which comes in two versions that are freely available for download,[9] as well as being accessible via the API of args.me. Version 1 of the corpus contains 387,606 arguments crawled from four debate portals in the middle of 2019 (debatewise.org, idebate.org, debatepedia.org, and debate.org). Each argument in the corpus consists of a conclusion (a claim that something is true or favorable) and a text covering one or more premises (reasons supporting or attacking the claim). Version 2 contains 387,740 arguments crawled from the same four debate portals, and 48 arguments from Canadian parliament discussions. In addition to the conclusions and premises, Version 2 also includes the texts surrounding them on the pages from which the arguments were extracted.

[8] https://www.args.me/api-en.html

[9] https://webis.de/data.html#args-me-corpus

Table 2. Overview of the participants' strategies for Task 1. Swordsman serves as a baseline using Lucene's DirichletLM retrieval model. Teams are ordered descending by nDCG@5.

Team	Retrieval	Augmentation	(Re)ranking Feature
Dread Pirate Roberts	DirichletLM/Similarity-based	Language modeling	—
Weiss Schnee	DPH	Embeddings	Quality
Prince of Persia	Multiple models	Synonyms	Sentiment
The Three Mouseketeers	DirichletLM	—	—
Swordsman (Baseline)	DirichletLM	—	—
Thongor	BM25/DirichletLM	—	—
Oscar François de Jarjayes	DPH/Similarity-based	—	Sentiment
Black Knight	TF-IDF	Cluster-based	Stance, readability
Utena Tenjou	BM25	—	—
Arya Stark	BM25	—	—
Don Quixote	Divergence from Randomness	Cluster-based	Quality + Similarity
Boromir	Similarity-based	Topic modeling	Author credibility
Aragorn	BM25	—	Premise prediction
Zorro	BM25	—	Quality + NER

4.3 Survey of Submissions to Task 1

The submissions to Task 1 largely employ a similar general strategy, which is characterized by the following three components: (1) a retrieval strategy; (2) an augmentation component, where either the query set is expanded, or results are extended directly based on features of documents in an initially retrieved set; (3) a (re)ranking component based on a primary document feature, which weighs, boosts, or modifies the retrieval scores, or which is used directly to rank the results.

In Table 2, we provide a high-level overview of the participants' systems. Each run is characterized with regard to the three aforementioned components. For their retrieval component, most teams opted for one of the four models also evaluated for argument search by Potthast et al. [31]. In line with the authors' results, approaches using DirichletLM or DPH seem to be far better than approaches relying on BM25 or TF-IDF. Two teams further rely on the cosine similarity, which yields favorable results as well.

Six teams opted to integrate query or result augmentation into their pipeline. Most notably, the three top-performing approaches use different ways of query augmentation, aimed either at generating synonymous queries, or at generating new queries from scratch using large language models. Others use result augmentation, opting for a cluster-based approach to group arguments based on text-inherent features, such as topic models or semantic clustering. In an effort to increase the topical coverage of results for a single query, arguments belonging to a cluster present in the initially retrieved results are returned as well—even though they may not be directly related to the query. Query augmentation seems to be more successful than direct result augmentation.

For re-ranking, we can differentiate two main types of features being chosen: three teams integrate the notion of argument quality into their ranking process; two other teams utilize sentiment analysis. The first choice operates under the hypothesis that an argument of higher quality will likely be of higher relevance as well. For the second choice, one team proposes that a neutral sentiment coincides with higher relevance, while the other argues that a high sentiment value hints at an emotionally involved author, possibly arguing more convincingly.

Different other text-based re-ranking features were proposed as well, such as premise prediction scores, readability, or the presence of named entities. However, they all have only limited effects on performance. One team integrates external information by calculating credibility scores for authors of arguments in the corpus. Approaches relying on trained models, i.e., quality prediction and premise prediction scoring, perform less favorable in general, which may be due to the limited amount of domain-specific training data. It remains to be seen whether the results improve in future iterations of the Touché lab. In what follows, we first summarize the systems presented in a submitted paper, to provide more in-depth insights into the participants' approaches. We also received runs from six teams who did not submit a respective paper. Based on a consultation with those teams, we describe their systems in the latter part of this section.

Submissions with working notebooks

Aragorn by Entezari and Völske [13] employs a distant-supervision approach to train seven deep-learning models, the idea being that the argument retrieval task is similar to the structure of arguments in the args.me corpus: Retrieving relevant arguments for a query is comparable to retrieving premises that support an argument's conclusion. Models receive the conclusion as a query during the training process and rank possible premises, with premise/conclusion labels derived from the corpus. To construct training samples, the premise text of each argument is marked as relevant, whereas the top 100 premises from other arguments ranked by BM25 are marked as irrelevant. Using that methodology, a distant-supervision dataset is produced comprising over 300,000 training and 4,800 validation queries to train the seven deep neural models. The training comprises multiple epochs per model, using only the best-performing candidate per model for the actual argument retrieval task, according to MAP@20 on the validation queries. All seven trained models are combined using linear regression optimized on the validation queries. A run was submitted for this combined approach (Run 3), with additional individual runs for the four most promising trained models (Runs 1, 2, 4, and 5). For all five runs, the retrieval pipeline used BM25 to retrieve 100 arguments and then re-ranked them by the predicted model score.

Don Quixote by Dumani and Schenkel [12] follows a two-step approach: In an initial offline operation, the dataset is clustered based on Sentence-BERT (SBERT) embeddings of both the conclusions and the premises. Premises are further grouped by identical conclusions. This grouping allows to calculate the so-called dimension convincing frequencies (DCFs) of a premise, in comparison to other premises of the same conclusion. For the three quality dimensions cogency, reasonableness, and effectiveness, a logistic regression is applied to all possible conclusion-premise pairs. The DCFs are then obtained by counting how often a premise was better than other premises belonging to the same conclusion in a cross comparison. Conclusions and premises are indexed separately. At retrieval time, a set of premises is retrieved using a divergence-from-randomness model. This set is then extended with all other premises belonging to the same group. For each entry in the extended set, a score is calculated using both the similarity of the query to the conclusion and the sum of the three DCFs per premise. For

each group in this result set, the scores of its members are aggregated and a final ranking is obtained by choosing a single representative per group, and ranking representatives by group score. Representatives are chosen by text length, under the hypothesis that a longer premise is also more specific and therefore may be better suited.

Dread Pirate Roberts by Akiki and Potthast [3] employs transformer-based models as part of its argument retrieval pipeline, pursuing three independent approaches: (1) The initial query is expanded using GPT-2; by adding question-like suffixes to the query, the text generation is steered towards argumentative text. Thus, a set of queries is built from generated sentences, and for each, results are retrieved using a DirichletLM retrieval model. Finally, all results are combined and ranked by their respective absolute score (Run 1). (2) Similar to the first approach, query expansion is achieved by generating argumentative text. However, instead of generating sentences, single word predictions by BERT are used. Once again, question-like suffixes are employed to influence the nature of the generated text. Final results are obtained by composing a query out of the single generated terms and retrieving results with a DirichletLM model (Run 2). (3) Instead of focusing on query expansion, the third approach uses a transformer-based model to obtain document representations. Arguments are embedded in a vector space using Google's BERT-like Universal Sentence Encoder (USE). Retrieval is subsequently conducted using nearest-neighbor search with respect to the query (Runs 3, 4, and 5).

Oscar François de Jarjayes by Staudte and Lange [41] combines the traditional DPH retrieval model with document similarities based on a CBOW dual embedding space model. The intersection of the top-1000 result sets of both retrieval strategies is ranked descending by DPH score, producing a search result that ensures contextual relevance, as determined by the dual embedding space model, as well as query-specific accuracy, as given by the DPH scoring. Furthermore, a sentiment-based score weighting is proposed under the hypothesis that texts written by emotionally involved authors (high sentiment values according to the Google Cloud Natural Language API) are of higher quality and relevance to a query than neutral documents. An evaluation based on the retrieval performance of different sentiment weighting schemes seems to support this hypothesis.

Weiss Schnee by Bundesmann et al. [24] integrate a notion of argument quality as well as result heterogeneity into the argument retrieval process. First, quality ratings for all arguments in the args.me corpus are predicted using Support Vector Regression (SVR) based on 22 text features. For retrieval, the authors considered three different strategies of query expansion, one based on WordNet synonyms, the other two using embedding-based language modeling. Based on the augmented queries, an initial set of arguments is retrieved and scored by a DPH retrieval model, additionally weighted by their quality score. The top-8 results are then re-ranked, maximizing the weighted sum of semantic distance between consecutive entries in the result list, thus ensuring result heterogeneity, as well as each entries' argumentative quality. However, this re-ranking approach is considered an experimental feature and is not included in officially submitted runs, as the nDCG evaluation measure does not take diversity of results into account.

Zorro by Shahshahani and Kamps [38] was evaluated with only a single run. First, document encodings are constructed using BERT on the first 512 tokens of each argument in the args.me corpus. A ranking is then created in three steps. (1) An initial ranking is produced using BM25. (2) An SVM classifier trained on the Dagstuhl-15512 ArgQuality corpus [44] is used to predict scores for cogency, well-writtenness, reasonableness and overall quality of each argument in the initial ranking. (3) The final ranking is then produced using the learning-to-rank library RankLib. In addition to the scores predicted in the previous step, training incorporates the following features: the BERT encoding, two named entity-based features, and two binary features indicating the presence of numerical named entities (percent, quantity, money) and other entities (person, location, organization) in the argument. The assumption is that an argument exhibiting such entities is more likely to provide users with persuasive and effective information.

Submissions without working notebooks

Black Knight clusters the arguments in the corpus based on their debate titles. A query is then associated to a cluster, and all arguments belonging to the cluster are added to the result set. A ranking is obtained by calculating the cosine similarity between TF-IDF vectors of the arguments and the query. Moreover, a filtering pipeline is applied to ensure suitable length, readability, and stance of results. The team includes an extensive stance classification approach as part of their indexing.

Prince of Persia builds two separate indices for the premise and conclusion of each argument, the hypothesis being that due to the different lengths of both fields, using different retrieval models for each yields improved results overall. In five runs, different combinations of retrieval models are explored based on language models (DirichletLM), divergence from randomness (DLH) and term frequency (BM25) from the Terrier library [26]. Run 1 used DirichletLM for both indices, Run 2 used BM25. In Run 3, DLH and PL2 are combined, in Run 4 DLH and BM25, and in Run 5 DLH is used for both. Further a WordNet-based query augmentation is incorporated, and score weighting using sentiment analysis, boosting arguments with neutral sentiment in the ranking.

The Three Mouseketeers tackles the task of argument retrieval with a parameter optimization of Terrier's [26] implementation of the DirichletLM retrieval model, using pseudo-relevance feedback.

Boromir combines a word embedding space and a topic model to retrieve arguments that are both semantically related to the query as well as contextually related to each other. First, the cosine similarity is calculated on 300-dimensional word embeddings and used to retrieve an initial set of arguments. This set is extended with other arguments close to the initial results in the topic space, which is constructed using an LDA topic model with 80 topics. Furthermore, arguments are re-ranked based on author credibility: author statistics were crawled from the source page (if available) for each argument in the corpus. On this basis, an Elo rating was established for authors based on their debate winning statistics.

Table 3. Results for Task 1 on conversational argument retrieval for (a) args.me corpus Version 1, and (b) args.me corpus Version 2. The baseline approach is in bold.

(a)

Team	**nDCG@5**	**Team** (continued)	**nDCG@5**
Weiss Schnee	0.804	Prince of Persia (Run 4)	0.641
Prince of Persia (Run 1)	0.791	Don Quixote	0.617
The Three Mouseketeers	0.789	Aragorn (Run 3)	0.593
Swordsman (Baseline)	**0.769**	Prince of Persia (Run 5)	0.555
Dread Pirate Roberts (Run 2)	0.743	Aragorn (Run 2)	0.331
Prince of Persia (Run 2)	0.724	Aragorn (Run 4)	0.319
Thongor	0.717	Aragorn (Run 1)	0.288
Oscar François de Jarjayes	0.699	Aragorn (Run 5)	0.271
Black Knight	0.692	Boromir	0.152
Utena Tenjou	0.689		
Arya Stark	0.662		
Prince of Persia (Run 3)	0.642		

(b)

Team	**nDCG@5**
Dread Pirate Roberts (Run 1)	0.808
Swordsman (Baseline)	**0.756**
Dread Pirate Roberts (Run 2)	0.755
Aragorn (Run 1)	0.684
Dread Pirate Roberts (Run 3)	0.598
Zorro	0.573
Dread Pirate Roberts (Run 4)	0.527
Dread Pirate Roberts (Run 5)	0.519
Aragorn (Run 2)	0.404
Aragorn (Run 3)	0.372
Aragorn (Run 4)	0.371
Aragorn (Run 5)	0.319

Arya Stark and Thongor uses the BM25 model and preprocesses the queries by trying different combinations of the tokens (shingling). Before generating combinations for a topic, stop words are removed and the tokens are stemmed.

Utena Tenjou creates two indices for the conclusion and premise and weights these two fields differently. In different runs, the weight of the premise is fixed at 1, adjusting the weight of the conclusion between 0.40 and 3.65. The same weighting mechanism is applied using two different retrieval models: BM25 and DirichletLM.

4.4 Task Evaluation

In this first edition of the lab, we evaluated only the *relevance* of the retrieved documents (not the quality of the comprised arguments), given that the collection of manual judgments is a rather complex and time-consuming task. We collected the participants' results as classical TREC-style runs where, for each topic, the document IDs are returned in a ranked list, ordered by descending relevance (i.e., the most relevant document should occur at Rank 1). The document pools for judgments were created with the trectools Python library [28],[10] using a top-5 pooling strategy that resulted in 5,291 unique retrieval results to be judged.

The relevance judgments were collected on Amazon Mechanical Turk, following previously designed annotation guidelines [14, 31]. We tasked the crowdworkers to decide whether or not a given retrieved text is an argument, and to annotate the relevance of the item on a scale ranging from 1 (low) to 5 (high). Non-arguments were subsequently marked as spam and received a score of –2. Each retrieval result was separately annotated by five crowdworkers, using majority vote as a decision rule. To further ensure the annotation quality, we recruited only workers with an approval rate of at least 95%, and checked for occurrences of systematic spam.

In total, 2,964 relevance judgments were collected for Version 1 of the args.me corpus, and 2,298 relevance judgments were collected for Version 2. Due to an annotation error, no judgments were collected for topic 25, prompting us to omit this topic from further evaluation. Notice that this reduces the total number of retrieval results in pooling

[10] https://pypi.org/project/trectools/

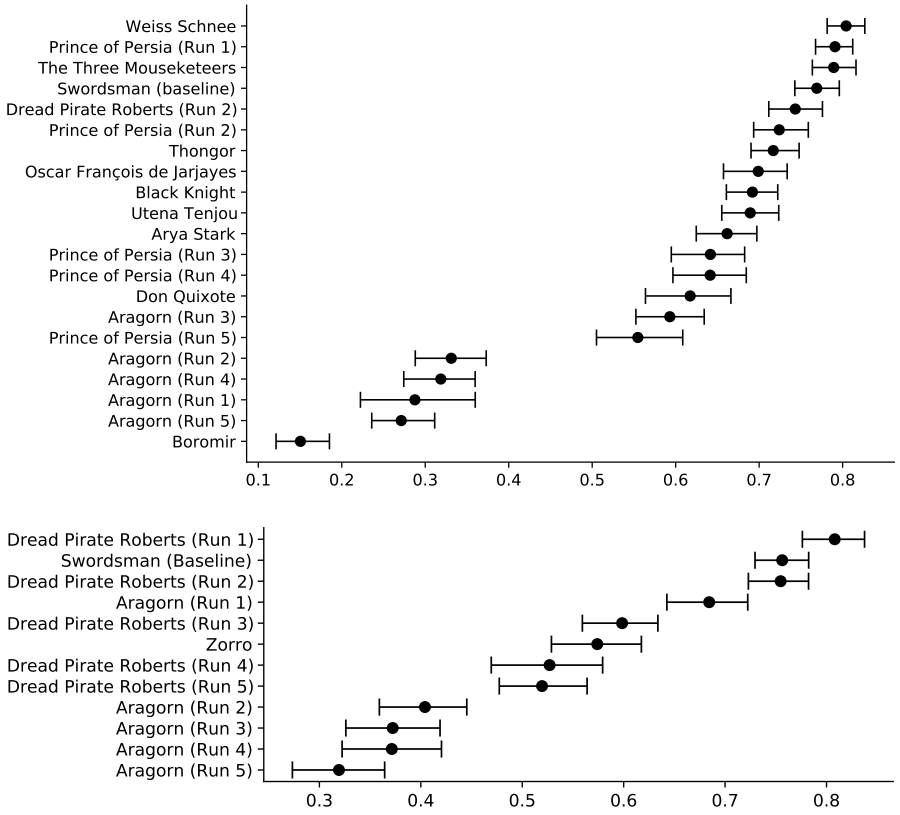

Figure 1. Mean nDCG@5 and 95% confidence intervals for runs submitted to Task 1, args.me corpus Version 1 (top), and Version 2 (bottom).

from 5,291 to a total of 5,262. The participants' approaches were subsequently evaluated using nDCG [17] with an evaluation depth of 5 on the remaining 49 topics. We used the nDCG implementation provided by the trec_eval library.[11] Results are given in Table 3a and b for the first and second version of the dataset, respectively. Additionally, in Figure 1, the plots display effectiveness and 95% confidence intervals for each run on the first and second version of the args.me dataset, respectively. Confidence intervals were obtained using bootstrapping ($n = 10,000$).

As baseline run, the args.me corpus was indexed in Elasticsearch and results for each topic were retrieved using the Lucene implementation of the DirichletLM model [48].[12] This retrieval model has shown favorable performance in prior experiments on argument search [31]. No additional query augmentation or re-ranking strategies were used.

[11] https://trec.nist.gov/trec_eval/

[12] https://lucene.apache.org/core/8_6_0/core/org/apache/lucene/search/similarities/LMDirichletSimilarity.html

5 Touché Task 2: Comparative Argument Retrieval

The goal of the Touché's second task is to support individuals' personal decisions in everyday life that can be expressed as a comparative question (e.g., "Is X better than Y with respect to Z?") and that do not have a single "factual" answer. Such questions can, for instance, be found on community question answering (CQA) sites like Yahoo! Answers or Quora, or in discussions on Reddit, but are also submitted as queries to search engines. The search engines then often simply show content from CQA websites or some web document mentioning the query terms as a direct answer above the classic "ten blue links." However, a problem of such attempts at short direct answers is that CQA websites may not always provide a diverse and sufficient overview of all possible options with well-formulated arguments, nor will all underlying textual information be credible—a broader set of such issues also forms the dilemma of direct answers [33]. As a first step to work on technology to present several credible arguments and different angles in potential direct comparative answers, Task 2 deals with the scenario of retrieving comparative arguments from web-scale collections.

5.1 Task Definition

The participants of Task 2 are asked to retrieve and rank documents from the ClueWeb12[13] that help to answer a comparative question. Ideally, the retrieved documents contain convincing arguments for or against some of the possible options for a given comparison. Similar to Task 1, participation is possible without indexing the document collection on the participants' side, since we provide easy access to the document collection through the BM25F-based ChatNoir search engine [6][14]—via a web-interface and an API. To identify arguments in texts, the participants are not restricted to any system; they can use own technology or any existing argument tagger of their choice. To lower the entry barriers for participants new to argument mining, we offer support via the neural argument tagger TARGER [9], hosted on our servers.

5.2 Data Description

Retrieval Topics. We selected 50 comparative questions from questions submitted to commercial search engines or asked on question answering platforms [7], each covering some personal decision from everyday life. For every question, we have formulated a respective TREC-style topic with the question as the title, a description of the searcher's possible context and information need, and a narrative describing what makes a result relevant (i.e., serving as a guideline for human assessors). An example topic is shown in Table 4. In the topic creation, we ensured through manual spot checks that the ClueWeb12 collection actually contains possibly relevant documents for all topics.

[13] https://lemurproject.org/clueweb12/

[14] https://www.chatnoir.eu/

Table 4. Example topic for Task 2 on comparative argument retrieval.

Number	16
Title	Should I buy or rent?
Description	A person is planning to move out from their current small flat to start a family. Hoping that the new family will stay together in the new place for some longer time, the person is considering to even buy a new home and not just to rent it. However, this is kind of an important decision with many different angles to be considered: financial situation, the duties coming with owning a flat/house, potential happiness living in a property owned by someone else without any further (financial) responsibilities when major redos are needed, etc.
Narrative	Highly relevant documents contain various pros and cons for buying or renting a home. Particularly interesting could be checklists of what to favor in what situations. Documents containing definitions and "smaller" comparisons of buying or renting a property are relevant. Documents without any personal opinion/recommendation or pros/cons are not relevant.

Document Collection. Task 2 is based on the ClueWeb12 crawl[15] from between February and May 2012 (733 million English web pages; 27.3TB uncompressed). Participants of Task 2 could index the ClueWeb12 on their own or could use the Elasticsearch-based ChatNoir API for a BM25F-based baseline retrieval.

5.3 Survey of Submissions to Task 2

Five teams submitted a total of eleven approaches to this task (ten of which plus the additional ChatNoir baseline are used to create the assessment pools). All approaches use the BM25F-based search engine ChatNoir [6] to retrieve candidate documents that are then re-ranked using machine learning models of different complexity in basically three steps: (1) Represent documents and queries using language models, (2) identify arguments and comparative structures in documents, and (3) assess argument quality. Only two approaches use query expansion techniques for retrieving the candidates before re-ranking. The characteristics of the teams' approaches are summarized in Table 5 and detailed below (teams ordered alphabetically).

Bilbo Baggins by Abye et al. [1] uses a two-stage retrieval pipeline: (1) Query expansion to increase the recall of the candidate retrieval, and (2) re-ranking the candidate documents using three feature types: relevance, credibility, and support features. Before querying ChatNoir, Bilbo Baggins expands topic titles with synonyms and antonyms from WordNet for entities (e.g., laptop and desktop) and comparison aspects (e.g., better, best) detected with Spacy.[16] Then, ChatNoir is queried with four queries for each topic: (1) the original topic title, (2) all identified entities as one conjunctive AND-query, (3) all entities and comparison aspects as one disjunctive OR-query, (4) all entities, aspects, their synonyms and antonyms as one disjunctive OR-query. The set of

[15] https://lemurproject.org/clueweb12/

[16] https://spacy.io/

Table 5. Overview of the participating teams' strategies for Task 2. All approaches use the BM25F-based search engine ChatNoir for an initial candidates retrieval (Puss in Boots as the baseline being the unchanged ChatNoir results).

Team	Representation	Query processing	(Re-)Ranking features
Bilbo Baggins	Bag of words	Named entities, comp. aspects	Credibility, support
Frodo Baggins	Bag of words	GloVe nearest neighbors	Simil. with gen. documents (GPT-2)
Inigo Montoya	Bag of words	Tokens & logic. OR	Argum. units (TARGER)
Katana	Diff. language models	Diff. language models	Comparativeness score
Puss in Boots	Bag of words	—	BM25F, SpamRank
Zorro	Bag of words	—	PageRank, argumentativeness

the top-30 results of each of these four queries are then re-ranked using "relevance features" (PageRank, number of comparative sentences as identified by an XGBoost classifier with InferSent embeddings [30], argument ratio Reimers et al. [34]), document "credibility" (SpamRank, BlocklistedLinks), and "support" features (number of sentences that support claims [35]), where features are respective numerical scores. The final ranking is created over the sums of the scores multiplied with weighting values.

Frodo Baggins by Sievers [39] explores the hypothesis that large language models, given a search query as input, can generate prototypical candidate documents similar to relevant documents. This prototype document is generated using the GPT-2 model conditioned on the original query and a maximum of 1024 tokens for the generated text. The TF-IDF-based cosine similarity between a ChatNoir search result document and the generated prototype document induces Frodo Baggins' re-ranking. As for query expansion, each term of the original query is augmented by its nearest neighbor according to the cosine similarity of GloVe embeddings.

Inigo Montoya by Huck [16] uses the topic titles as queries, retrieving ChatNoir's top-20 results. For each result, the TARGER argument tagger is used to extract the argumentative units (premises and claims) into a new document for each original result. These new documents are then BM25-indexed using the Whoosh python library[17](BM25 parameters b=0.75 and k_1=1.2, document body: the set of arguments of the original document, document title: document ID from the ClueWeb12). This index is then queried with the topic titles as a disjunctive OR-query.

Katana by Chekalina and Panchenko [8] comprises a total of seven different approaches (runs), which all re-rank ChatNoir's top-1000 results for the topic title as the query. The re-ranking is based on different language models to encode query-document pairs and different variations of similarity:

- Run 1 only removes "duplicate" results (based on an exact title match against higher-ranked results) from ChatNoir's original top-1000 results.
- For Run 2 and other following Runs, Katana re-ranks Run 1 (basically, ChatNoir's results) by using as the sorting criterion a document's original ChatNoir relevance score multiplied with a "comparativeness" score. This comparativeness score is

[17]https://pypi.org/project/Whoosh/

the number of comparative sentences in the document as identified by an XGBoost classifier with InferSent embeddings [30], which is additionally increased if a query's comparison objects, aspects, and predicates are found in a document (using a one-layer LSTM with 200 hidden units and BERT embeddings for the input, pre-trained on a dataset created by the authors).
- For Run 3, the final ranking is based on the cosine similarity between a query and a document using the ULMFiT language model [15] multiplied with the comparativeness score also used in Run 2.
- For Run 4, the final ranking is based on the cosine similarity between BERT encodings of the query and the document titles (document body neglected).
- For Run 5, the final ranking is based on a similarity score calculated using a complex formula composed of the weights from selected BERT attention heads in a standard transformer (cf. the team's paper [8] for more details).
- For Run 6, the final ranking is based on the cosine similarity between a query and a document using the ULMFiT language model (i.e., Run 3 without multiplication).
- For Run 7, the similarity score is an average of the scores of the other runs multiplied with the comparativeness score. Note that this run is not part of the assessment pool due to our human assessors' workload.

Puss in Boots is the baseline for Task 2, which simply uses the results that ChatNoir [6] returns for a topic's title. ChatNoir is an Elasticsearch-based search engine, indexing the complete ClueWeb12 (and also other web collections) by processing raw HTML documents using main content extraction, language detection, and extraction of metadata (keywords, headings, hostnames, etc.). During retrieval, ChatNoir combines BM25 scores of multiple fields (title, keywords, main content, and the full document) and uses the documents' SpamRank [10] as a threshold to remove spam.

Zorro by Shahshahani and Kamps [38] re-ranks ChatNoir's top-1000 results in three consecutive steps: (1) Non-overlapping subsets of 10 documents are re-ranked by descending PageRank scores (i.e., original ranks 1–10, ranks 11–20, etc.). (2) In non-overlapping subsets of 10 documents already re-ranked by the first step, documents from blogs and discussions (categorization based on the URL domain) are moved to the top (i.e., rank 20 after Step 1 might at most move up to rank 11). (3) In non-overlapping subsets of 20 documents re-ranked with Steps 1 and 2, documents are moved to the top that are more argumentative according to an SVM-based argumentativeness classifier. To train the classifier, team Zorro sampled 3000 documents from the args.me corpus [2] as positive examples (argumentative) and 3000 documents from the ClueWeb12 as negative examples (not argumentative) by collecting the at most top-100 results from the args.me and ChatNoir APIs when queried with the 50 topics from Task 1.

5.4 Task Evaluation

Similar to Task 1, in the first lab year, we only evaluate the relevance of the retrieved documents but not any other argument quality dimensions. Using a top-5 pooling strategy of the submitted runs, including the baseline, a total of 1,783 unique results were

Table 6. Results for Task 2 on comparative argument retrieval. Baseline approach is in bold.

Team	**nDCG@5**	**Team** (continued)	**nDCG@5**
Bilbo Baggins	0.580	Frodo Baggins	0.450
Puss in Boots (ChatNoir)	**0.568**	Zorro	0.446
Inigo Montoya	0.567	Katana (Run 4)	0.404
Katana (Run 1)	0.564	Katana (Run 5)	0.223
Katana (Run 2)	0.553	Katana (Run 6)	0.200
Katana (Run 3)	0.464		

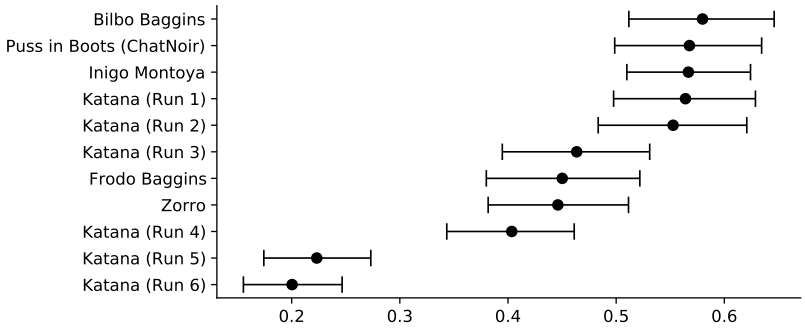

Figure 2. Mean nDCG@5 and 95% confidence intervals for runs submitted to Task 2.

judged by human assessors. To this end, we recruited seven graduate and undergraduate student volunteers, all with a computer science background. We used a κ-test of five documents from five topics to calibrate the annotators' interpretations of the guidelines (i.e., the topics including the narratives) and the three relevance labels: 0 (not relevant), 1 (relevant), and 2 (highly relevant). The original Fleiss' κ of 0.46 indicates a moderate agreement such that a follow-up discussion among the annotators was invoked to adjust their individual interpretations and to emphasize that documents should not be judged as highly relevant when they do not provide well-formulated evidence support. After the κ test, each annotator judged the results for disjoint subsets of the topics (i.e., each topic judged by one annotator only).

The achieved average nDCG@5 scores of the individual runs are given in Table 6, while Figure 2 also shows the 95% confidence intervals obtained using bootstrapping ($n = 10,000$). Only Bilbo Baggins achieves a slightly better average nDCG@5 score than the ChatNoir baseline by using query expansion and taking credibility and argumentativeness into account in the re-ranking. A reason for the pretty strong baseline effectiveness of Puss in Boots might be that, during the topic selection, the topic titles were already checked against the ClueWeb12 collection—and to this end the topic creators actually did submit the topic titles to ChatNoir and checked whether some result snippets at least contain the comparison items. However, since all the teams' approaches submitted to Task 2 do use the ChatNoir results as their candidates for re-ranking, the potential "easiness" of the topics did not really favor any approach. Still, topic selection is an issue that we will treat differently in the future.

The top-5 approaches (including the baseline) all lie in the 0.55-0.58 range with their average nDCG@5 scores. Interestingly, the top-4 runs are classical feature engineering approaches, while four out of the six lower-ranked runs (right side of Table 6) use deep learning-based language models. This observed difference in the effectiveness between the feature-based and the deep learning-based approaches might be caused by the fact that no training data was available, such that it will be interesting to observe whether any fine-tuning based on the now created ground truth might help in the future. That some more respective research effort is justified is also indicated by the fact that none of the approaches actually substantially improved upon the baseline ranking, even though there still is quite some headroom towards "perfect" effectiveness.

6 Summary and Outlook

Touché and its two shared tasks have been designed with the goal to establish a collaborative platform for researchers in the field of argument retrieval. Starting from argument relevance and argument quality corpora, Touché is meant to provide tools for the submission and evaluation of retrieval approaches, and to organize collaboration events such as workshops. By providing argument retrieval baselines and APIs, also researchers new to the field may quickly start developing their own approaches.

The first edition of Touché featured two tasks: (1) conversational argument retrieval to support argumentation on socially important problems in dialogue or debate scenarios, and (2) comparative argument retrieval to support decision making on a personal level. In total, 17 teams submitted 41 different approaches that were evaluated with respect to relevance. Still, we find that relatively "simple" argumentation-agnostic baselines like DirichletLM-based or BM25F-based retrieval are not substantially worse then the best approaches, or they are even are on a par with them. The best approaches share some common techniques such as query expansion, argument quality assessment, and the identification of comparative textual features in documents, but there still seems to be a lot of room for improvement. Further research on argument retrieval thus seems well-justified.

In the future, the participants will be able to use this year's relevance judgments to develop and fine-tune new approaches. We also plan to have deeper judgment pools and to additionally evaluate argument quality dimensions, such as logical cogency and strength of support.

Acknowledgments

We are very grateful to the CLEF 2020 organizers and the Touché participants, who allowed this lab to happen. We also want to thank our volunteer annotators who helped to create the relevance assessments.

This work was supported by the DFG through the project "ACQuA: Answering Comparative Questions with Arguments" (grants BI 1544/7-1 and HA 5851/2-1) as part of the priority program "RATIO: Robust Argumentation Machines" (SPP 1999).

Bibliography

[1] Abye, T., Sager, T., Triebel, A.J.: An Open-Domain Web Search Engine for Answering Comparative Questions—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)

[2] Ajjour, Y., Wachsmuth, H., Kiesel, J., Potthast, M., Hagen, M., Stein, B.: Data Acquisition for Argument Search: The args.me Corpus. In: Proceedings of the 42nd German Conference AI, KI 2019, Lecture Notes in Computer Science, vol. 11793, pp. 48–59, Springer (2019), https://doi.org/10.1007/978-3-030-30179-8_4

[3] Akiki, C., Potthast, M.: Exploring Argument Retrieval with Transformers—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)

[4] Aristotle, Kennedy, G.A.: On Rhetoric: A Theory of Civic Discourse. Oxford: Oxford University Press (2006)

[5] Bar-Haim, R., Krieger, D., Toledo-Ronen, O., Edelstein, L., Bilu, Y., Halfon, A., Katz, Y., Menczel, A., Aharonov, R., Slonim, N.: From Surrogacy to Adoption; From Bitcoin to Cryptocurrency: Debate Topic Expansion. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, pp. 977–990, Association for Computational Linguistics (2019), https://doi.org/10.18653/v1/p19-1094

[6] Bevendorff, J., Stein, B., Hagen, M., Potthast, M.: Elastic ChatNoir: Search Engine for the ClueWeb and the Common Crawl. In: Proceedings of the 40th European Conference on IR Research, ECIR 2018, Lecture Notes in Computer Science, vol. 10772, pp. 820–824, Springer (2018), https://doi.org/10.1007/978-3-319-76941-7_83

[7] Bondarenko, A., Braslavski, P., Völske, M., Aly, R., Fröbe, M., Panchenko, A., Biemann, C., Stein, B., Hagen, M.: Comparative Web Search Questions. In: Proceedings of the 13th International Conference on Web Search and Data Mining, WSDM 2020, pp. 52–60, ACM (2020), https://doi.org/10.1145/3336191.3371848

[8] Chekalina, V., Panchenko, A.: Retrieving Comparative Arguments using Deep Pre-trained Language Models and NLU—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)

[9] Chernodub, A.N., Oliynyk, O., Heidenreich, P., Bondarenko, A., Hagen, M., Biemann, C., Panchenko, A.: TARGER: Neural Argument Mining at Your Fingertips. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, pp. 195–200, Association for Computational Linguistics (2019), https://doi.org/10.18653/v1/p19-3031

[10] Cormack, G., Smucker, M., Clarke, C.: Efficient and effective spam filtering and re-ranking for large web datasets. Information retrieval **14**(5), 441–465 (2011)

[11] Dumani, L., Neumann, P.J., Schenkel, R.: A Framework for Argument Retrieval - Ranking Argument Clusters by Frequency and Specificity. In: In Proceedings of the 42nd European Conference on IR Research, ECIR 2020, Lecture Notes in Computer Science, vol. 12035, pp. 431–445, Springer (2020), https://doi.org/10.1007/978-3-030-45439-5_29

[12] Dumani, L., Schenkel, R.: Ranking Arguments by Combining Claim Similarity and Argument Quality Dimensions—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)

[13] Entezari, S., Völske, M.: Argument Retrieval Using Deep Neural Ranking Models—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)

[14] 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, ACL 2020, pp. 5772–5781, Association for Computational Linguistics (2020), URL https://www.aclweb.org/anthology/2020.acl-main.511/

[15] Howard, J., Ruder, S.: Universal Language Model Fine-tuning for Text Classification. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, pp. 328–339, Association for Computational Linguistics (2018), https://doi.org/10.18653/v1/P18-1031
[16] Huck, J.: Development of a Search Engine to Answer Comparative Queries—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)
[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] Jindal, N., Liu, B.: Identifying Comparative Sentences in Text Documents. In: Proceedings of the 29th Annual International Conference on Research and Development in Information Retrieval, SIGIR 2006, pp. 244–251, ACM (2006), https://doi.org/10.1145/1148170.1148215
[19] Jindal, N., Liu, B.: Mining Comparative Sentences and Relations. In: Proceedings of the 21st National Conference on Artificial Intelligence and the 18th Innovative Applications of Artificial Intelligence Conference, AAAI 2006, pp. 1331–1336, AAAI Press (2006), URL http://www.aaai.org/Library/AAAI/2006/aaai06-209.php
[20] Kessler, W., Kuhn, J.: A Corpus of Comparisons in Product Reviews. In: Proceedings of the 9th International Conference on Language Resources and Evaluation, LREC 2014, pp. 2242–2248, European Language Resources Association (ELRA) (2014), URL http://www.lrec-conf.org/proceedings/lrec2014/summaries/1001.html
[21] Levy, R., Bogin, B., Gretz, S., Aharonov, R., Slonim, N.: Towards an argumentative content search engine using weak supervision. In: Bender, E.M., Derczynski, L., Isabelle, P. (eds.) Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, pp. 2066–2081, Association for Computational Linguistics (2018), URL https://www.aclweb.org/anthology/C18-1176/
[22] 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, Association for Computational Linguistics (2020), URL https://www.aclweb.org/anthology/2020.acl-main.512/
[23] Mass, Y., Shechtman, S., Mordechay, M., Hoory, R., Shalom, O.S., Lev, G., Konopnicki, D.: Word Emphasis Prediction for Expressive Text to Speech. In: Proceedings of the 19th Annual Conference of the International Speech Communication Association, Interspeech 2018, pp. 2868–2872, ISCA (2018), https://doi.org/10.21437/Interspeech.2018-1159
[24] Bundesmann, M., Christ, L., Richter, M.: Creating an Argument Search Engine for Online Debates—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020), ISSN 1613-0073
[25] Nadamoto, A., Tanaka, K.: A Comparative Web Browser (CWB) for Browsing and Comparing Web Pages. In: Proceedings of the 12th International World Wide Web Conference, WWW 2003, pp. 727–735, ACM (2003), https://doi.org/10.1145/775152.775254
[26] Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C., Johnson, D.: Terrier information retrieval platform. In: Advances in Information Retrieval, 27th European Conference on IR Research, ECIR 2005, Proceedings, Lecture Notes in Computer Science, vol. 3408, pp. 517–519, Springer (2005), https://doi.org/10.1007/978-3-540-31865-1_37,
[27] Page, L., Brin, S., Motwani, R., Winograd, T.: The PageRank Citation Ranking: Bringing Order to the Web. Tech. rep., Stanford InfoLab (1998)
[28] Palotti, J.R.M., Scells, H., Zuccon, G.: TrecTools: an Open-source Python Library for Information Retrieval Practitioners Involved in TREC-like Campaigns. In: Proceedings of

the 42nd International Conference on Research and Development in Information Retrieval, SIGIR 2019, pp. 1325–1328, ACM (2019), https://doi.org/10.1145/3331184.3331399
[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, Association for Computational Linguistics (2019), https://doi.org/10.18653/v1/w19-4516
[30] Panchenko, A., Bondarenko, A., Franzek, M., Hagen, M., Biemann, C.: Categorizing Comparative Sentences. In: Stein, B., Wachsmuth, H. (eds.) 6th Workshop on Argument Mining (ArgMining 2019) at ACL, Association for Computational Linguistics (Aug 2019), URL https://www.aclweb.org/anthology/W19-4516
[31] Potthast, M., Gienapp, L., Euchner, F., Heilenkötter, N., Weidmann, N., Wachsmuth, H., Stein, B., Hagen, M.: Argument Search: Assessing Argument Relevance. In: Proceedings of the 42nd International Conference on Research and Development in Information Retrieval, SIGIR 2019, pp. 1117–1120, ACM (2019), https://doi.org/10.1145/3331184.3331327
[32] Potthast, M., Gollub, T., Wiegmann, M., Stein, B.: TIRA Integrated Research Architecture. In: Information Retrieval Evaluation in a Changing World - Lessons Learned from 20 Years of CLEF, The Information Retrieval Series, vol. 41, pp. 123–160, Springer (2019), https://doi.org/10.1007/978-3-030-22948-1_5
[33] Potthast, M., Hagen, M., Stein, B.: The Dilemma of the Direct Answer. SIGIR Forum **54**(1) (Jun 2020), URL http://sigir.org/forum/issues/june-2020/
[34] Reimers, N., Schiller, B., Beck, T., Daxenberger, J., Stab, C., Gurevych, I.: Classification and Clustering of Arguments with Contextualized Word Embeddings. In: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Volume 1: Long Papers, pp. 567–578, Association for Computational Linguistics (2019), https://doi.org/10.18653/v1/p19-1054
[35] Rinott, R., Dankin, L., Perez, C.A., 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, EMNLP 2015, pp. 440–450, The Association for Computational Linguistics (2015), https://doi.org/10.18653/v1/d15-1050
[36] Robertson, S.E., Zaragoza, H., Taylor, M.J.: Simple BM25 extension to multiple weighted fields. In: Grossman, D.A., Gravano, L., Zhai, C., Herzog, O., Evans, D.A. (eds.) Proceedings of the 2004 ACM CIKM International Conference on Information and Knowledge Management, Washington, DC, USA, November 8-13, 2004, pp. 42–49, ACM (2004), https://doi.org/10.1145/1031171.1031181
[37] Schildwächter, M., Bondarenko, A., Zenker, J., Hagen, M., Biemann, C., Panchenko, A.: Answering Comparative Questions: Better than Ten-Blue-Links? In: Proceedings of the Conference on Human Information Interaction and Retrieval, CHIIR 2019, pp. 361–365, ACM (2019), https://doi.org/10.1145/3295750.3298916
[38] Shahshahani, M.S., Kamps, J.: University of Amsterdam at CLEF 2020—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)
[39] Sievers, B.: Question Answering for Comparative Questions with GPT-2—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)
[40] Stab, C., Daxenberger, J., Stahlhut, C., Miller, T., Schiller, B., Tauchmann, C., Eger, S., Gurevych, I.: ArgumenText: Searching for Arguments in Heterogeneous Sources. In: Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2018, pp. 21–25, Association for Computational Linguistics (2018), https://doi.org/10.18653/v1/n18-5005

[41] Staudte, C., Lange, L.: SentArg: A Hybrid Doc2Vec/DPH Model with Sentiment Analysis Refinement—Notebook for the Touché Lab on Argument Retrieval at CLEF 2020. In: Working Notes Papers of the CLEF 2020 Evaluation Labs (Sep 2020)
[42] 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
[43] Wachsmuth, H., Naderi, N., Habernal, I., Hou, Y., Hirst, G., Gurevych, I., Stein, B.: Argumentation Quality Assessment: Theory vs. Practice. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, pp. 250–255, Association for Computational Linguistics (2017), https://doi.org/10.18653/v1/P17-2039
[44] Wachsmuth, H., Naderi, N., Hou, Y., Bilu, Y., Prabhakaran, V., Thijm, T.A., Hirst, G., Stein, B.: Computational argumentation quality assessment in natural language. In: Lapata, M., Blunsom, P., Koller, A. (eds.) Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Volume 1: Long Papers, pp. 176–187, Association for Computational Linguistics (2017), https://doi.org/10.18653/v1/e17-1017
[45] Wachsmuth, H., Potthast, M., Khatib, K.A., Ajjour, Y., Puschmann, J., Qu, J., Dorsch, J., Morari, V., Bevendorff, J., Stein, B.: Building an Argument Search Engine for the Web. In: Proceedings of the 4th Workshop on Argument Mining, ArgMining@EMNLP 2017, pp. 49–59, Association for Computational Linguistics (2017), https://doi.org/10.18653/v1/w17-5106
[46] Wachsmuth, H., Stein, B., Ajjour, Y.: "PageRank" for Argument Relevance. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, pp. 1117–1127, Association for Computational Linguistics (2017), https://doi.org/10.18653/v1/e17-1105
[47] Wachsmuth, H., Syed, S., Stein, B.: Retrieval of the Best Counterargument without Prior Topic Knowledge. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, pp. 241–251, Association for Computational Linguistics (2018), URL https://www.aclweb.org/anthology/P18-1023/
[48] Zhai, C., Lafferty, J.D.: A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 334–342, ACM (2001), https://doi.org/10.1145/383952.384019,