Quality-Aware Ranking of Arguments

Lorik Dumani
dumani@uni-trier.de
Trier University
Trier, Germany

Ralf Schenkel
schenkel@uni-trier.de
Trier University
Trier, Germany

ABSTRACT

Argument search engines identify, extract, and rank the most important arguments for and against a given controversial topic. A number of such systems have recently been developed, usually focusing on classic information retrieval ranking methods that are based on frequency information. An important aspect that has been ignored so far by search engines is the quality of arguments. We present a quality-aware ranking framework for arguments already extracted from texts and represented as argument graphs, considering multiple established quality measures. An extensive evaluation with a standard benchmark collection demonstrates that taking quality into account significantly helps to improve retrieval quality for argument search. We also publish a dataset in which arguments with respect to topics were tediously annotated by humans with three widely accepted argument quality dimensions.

CCS CONCEPTS

• **Information systems** → **Information retrieval**; *Information retrieval query processing*; *Retrieval models and ranking*; *Relevance assessment*.

KEYWORDS

argument retrieval, argument ranking, argument quality dimensions, probabilistic framework

ACM Reference Format:
Lorik Dumani and Ralf Schenkel. 2020. Quality-Aware Ranking of Arguments. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3340531.3411960

1 INTRODUCTION

Argumentation exists in many different variants and has always been used by people, for example, to convince others of certain views and maybe to persuade them to undertake certain actions. Alternatively, one can just use other people's arguments to form an opinion on an issue or topic.

A widely used definition for an *argument* describes it as a *claim* supported or attacked by at least one *premise* with an inference [35], which also already induces a simple graph structure. The claim is central and usually a controversial point of view, which should not be accepted by the reader immediately. This acceptance is typically increased or decreased only with the help of more information (by premises) [34]. Sometimes the terms premise and argument are used synonymously, which is because some define an argument as a premise along with the inference to the claim. An example for a claim could be "*measles vaccinations must be executed*", and an example for an supporting premise for this claim could be "*measles vaccinations reduce death rates*". Two premises have the same *stance* towards a claim if they both support or attack the claim.

Due to its size, practically all arguments on all topics can be found on the Web. However, existing search engines usually only retrieve documents containing them but cannot identify and rank the premises for a given claim. In contrast, dedicated argumentation search engines, which are heavily investigated, are expected to extract premises from documents, remove (near) duplicates and return the most important pro and con premises. Within the argumentation community, there are at least two large subcommunities: one deals with argument mining, i.e., the extraction of arguments from natural language texts and their transformation into graph structures [26]. A good overview of argument mining is provided by Cabrio and Villata [6]. The other subcommunity works with already existing graph structures and aims to provide the best supporting and attacking premises for a user query, applying methods from information retrieval. Wachsmuth et al. [36] and Stab et al. [33] already proposed prototypes of such argumentation search engines.

However, ranking premises is a particularly challenging task. It is not sufficient to consider only the text-based similarity of the premise to the query claim, as we have shown in our prior work [11]; instead, the similarity of the premise's original claim to the query claim plays an important role. In this paper, we now argue that the quality of the premise should also be taken into account. We will illustrate this idea using the convincingness of a premise, one of the quality dimensions often discussed for premises in the literature [14, 15, 37]. Consider again the claim "*measles vaccinations must be executed*" mentioned before. A logically conclusive premise such as "*a measles vaccination is important because of its high seroconversion rate*" to this claim may not be convincing to a non-expert user because the medical term "seroconversion rate" might not be well understood.[1] At the same time, a logically less conclusive premise such as "*the lives of the children are in extreme danger if they are not vaccinated*" can be more convincing to the same user. In addition to convincingness, the literature has introduced various *argument quality dimensions*; a good overview is provided by Wachsmuth et al. [37], we provide a brief summary in Section 2.

[1] Seroconversion describes the development of specific antibodies during an infection or a vaccination.

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-6859-9/20/10…$15.00
https://doi.org/10.1145/3340531.3411960

To the best of our knowledge, our prior work [10] introduced the first probabilistic framework for retrieving premises from already mined argument graphs, given a query claim. This framework operates on clusters of claims and clusters of premises where clusters are formed based on similarity. It then ranks premises similar to the principle of TF-IDF, i.e., a premise is ranked high if similar premises frequently support (or attack) similar claims, and similar premises support as few other claim clusters as possible. Although it outperforms current methods of existing argument search engines which use the similarity of query and premise for ranking, a weakness in this approach is that it works exclusively with frequencies and does not take argument quality into account that can have varying effects on different people.

In this paper we extend this probabilistic ranking framework to take argument quality into account when ranking premises. We show how to include different argument quality dimensions and evaluate their effect on retrieval performance. As baseline we use the framework of our previous work which works exclusively with frequencies. We use the corpus proposed by Ajjour et al. [1]. It consists of 387,606 arguments crawled from four different debate portals and is the underlying corpus of the argument search engine ARGS by Wachsmuth et al. [36]. We use 50 queries, covering many topics provided by the CLEF lab Touché [4]. The best ten results per query for the baseline and our approach are pooled, annotated with respect to three state-of-the-art argument quality dimensions by two annotators, and evaluated with nDCG [17]. Our results show that the baseline is significantly outperformed. To foster the research for ranking arguments, we publish this carefully labeled dataset together with our classifiers that decide for two premises which of the two is more favorable with respect to a certain quality dimension.[2] We also provide the API for the computations.

Next, we discuss related work in Section 2. Then we introduce the foundations of the probabilistic framework of our prior work [10] in Section 3 and extend it to include argument quality in Section 4. Afterwards, we evaluate our approach in Section 5.Section 6 concludes the paper and discusses ideas for future work.

2 RELATED WORK

Since our work addresses the retrieval and ranking of arguments, we will describe existing argument search engines here, and in particular address their ranking procedures. We also discuss the argument quality dimensions.

Probabilistic Framework for Argument Retrieval. In this paper we extend the probabilistic framework of our prior work [10], which clusters claims and premises according to their meaning and then ranks premises similar to the principle of TF-IDF to a query. A premise is ranked high if similar premises frequently support (or attack) similar claims, and similar premises support as few other claim clusters as possible. We describe this framework in detail in Section 3. We evaluated this approach on a private dataset, which is similar to ARGS [1] and consists of arguments from debate portals, and compare it to a baseline system proposed by Wachsmuth et al. [36] which ranks premises according to the BM25F scoring model [28]. For this we picked 30 claims on the topic energy from this corpus and use them as queries. We then evaluated the selection of a representative of the clusters (the longest premise in a cluster is chosen as representative) as well as clustering and significantly outperform the baseline in terms of a simplified variant of α-nDCG [7]. One reason for the better performance is that this approach considers clusters of premises with the same meaning and only the representatives from these are shown. Thus the information gain is higher than with repetitive premises. However, this approach has the weakness that it only works with frequencies of premises and claims. We will extend this framework to include argument quality and compare our extended approach to the original one.

Argument Search Engines. A number of existing argument search engines provide pro and con arguments for user queries. Wachsmuth et al. [36] present ARGS, Stab et al. [33] present ARGUMENTEXT. ARGS works with arguments crawled from five debate portals[3] indexed by the Java framework APACHE LUCENE[4]. For the ranking they use the BM25F scoring model [28], which indexes the premises together with their associated claims, giving the claims more importance. ARGUMENTEXT first finds relevant documents and then identifies relevant premises in these documents. Here ELASTICSEARCH[5] is used together with the scoring model OKAPI BM25 [27]. Our work is comparable to these systems in the sense that our extended framework operates on a given large body of arguments. In fact, we use the ARGS corpus [1].

Argument Quality. Wachsmuth et al. [37] provide a survey of work on argument quality. Additionally, they introduce a taxonomy consisting of 15 argument quality dimensions based on the work of Blair [3]. Here one wants to evaluate an argument according to its specific argument quality dimension, e.g., whether it is logically conclusive or simply convincing because of the emotions it provokes. Besides the overall argumentation quality there are three main quality dimensions (1) *logical quality* in terms of the *cogency* or strength of an argument, (2) *rhetorical quality* in terms of the persuasive effect of an argument or argumentation (called *effectiveness*), and (3) *dialectical quality* in terms of the *reasonableness* of argumentation for resolving issues. These dimensions can be further subdivided to give a total of 15 widely accepted quality sub-dimensions. We will focus on the three main dimensions in the evaluation and train classifiers for them.

Wachsmuth et al. [37] also provide a corpus for argument quality. Three experts have annotated the 15 different dimensions on a scale from 1 (low) to 3 (high) for 32 (issue,stance) pairs of 10 premises each. An example for an issue is "*is the school uniform a good or bad idea*" and the stances for this issue are "*good*" or "*bad*". Habernal and Gurevych [15] take a different approach. They rank arguments only by their convincingness compared to other arguments. There is no score and no specific dimension. In their work, they have about 16k pairs of arguments assessed by five crowdworkers each, which argument is better and used the dataset to train a feature-rich SVM and Bi-LSTM. Among others, they included uni- and bi-gram presence, ratio of adjective and adverb endings and many more as features. They report an accuracy between .76-.78.

[2]The data is available via the following link: https://basilika.uni-trier.de/nextcloud/s/XQ6pJbKPfKnDkU7/

[3]The crawled debate portals are idebate.org, debatepedia.org, debatewise.org, debate.org, and forandagainst.com.

[4]https://lucene.apache.org/

[5]https://www.elastic.co/

While the approach of Wachsmuth et al. [37] has the advantage of quality dimensional diversity, so that arguments can be ranked according to specific dimensions, because different dimensions have different effects on people, the approach of Habernal and Gurevych [15] has the advantage that a ranking can easily be extended when a new argument arrives, because a new argument can always have a better rating than the current best one which might already have the highest possible score. We extend the framework of our prior work [10] by quality dimensions, but we do not want to assign final scores. Therefore we use the dataset of Wachsmuth et al. [37], but do not learn any scores, but transform the dataset to (premise_1, premise_2) pairs and then learn which argument is better with regard to a specific dimension similar to the approach of Habernal and Gurevych [15]. This provides the added benefit of learning with more data.

3 A PROBABILISTIC FRAMEWORK FOR ARGUMENT RETRIEVAL

In this section we summarize the main components of the probabilistic ranking framework of our prior work [10] on which our method builds. This framework finds good premises for a query from a large corpus of already mined arguments using a two-step retrieval approach. Given a controversial *query claim*, the system first finds similar *result claims* in this corpus, following the intuition that the more similar a claim is to the query, the more relevant are the claim's premises to the query [11]. It then clusters and ranks all the associated *result premises* in the second step.

3.1 Probabilistic Framework

Given a large corpus of claims $C = \{c_1, c_2, \ldots\}$ and premises $\mathcal{P} = \{p_1, p_2, \ldots\}$, a set of disjoint claim clusters $\Gamma = \{\gamma_1, \gamma_2, \ldots\}$ is constructed, where each claim cluster γ_k consists of claims with the same meaning. Similarly, a set of disjoint premise clusters $\Pi = \{\pi_1, \pi_2, \ldots\}$ is constructed, each cluster π_j consisting of premises with the same meaning.

Let q be the query claim. The goal is to find the best clusters of supporting premises π^+ and the best clusters of attacking premises π^- for q. Since premises in a cluster have the same meaning, it is sufficient to pick only one representative per cluster. For simplicity, we will restrict the discussion to clusters of supporting premises; the definitions for clusters of attacking premises are analogous.

For ranking the premise clusters, one estimates the relevance probability $P(\pi^+|q)$ for all clusters $\pi^+ \in \Pi$ of supporting premises. To formalize this, we first consider single premises before we deal with clusters of premises.

Let $P(c|q)$ denote the probability that claim c is relevant to query q. Furthermore, let $P(p^+|c, q)$ denote the probability that a user selects premise p from c among all its supporting premises. The probability $P(p^+|q)$ that the user chooses the supporting premise p for the query claim q is now computed by summing $P(c|q) \cdot P(p^+|c, q)$ over all claims in the corpus:

$$P(p^+|q) = \sum_{c \in C} P(c|q) \cdot P(p^+|c, q) \tag{1}$$

where $\sum_{c \in C} P(c|q) = 1$. Since $P(p^+|c, q) = 0$ if p is not a premise of c we make the simplifying assumption that $P(p^+|c, q) = P(p^+|c)$. So the resulting formula is:

$$P(p^+|q) = \sum_{c \in C: p^+ \to c} P(c|q) \cdot P(p^+|c) \tag{2}$$

The relevance probability $P(\pi_j^+|q)$ for a cluster π_j^+ of supporting premises is finally calculated as the sum of all $P(p^+|q)$ for all premises $p^+ \in \pi_j^+$:

$$P(\pi_j^+|q) = \sum_{p^+ \in \pi_j^+} P(p^+|q) \tag{3}$$

3.2 Estimators for the Probabilities

Our previous work [10] provides estimators for the probabilities introduced above. $P(c|q)$ is estimated based on standard text retrieval methods; in this case Divergence from Randomness (DFR) [2], which yields the best results for claim retrieval.

Regarding the premises, the work proposes an approach similar to TF-IDF [30] where premises are ranked high that frequently support or attack claims in one claim cluster but rarely claims in other claim clusters. The probability $P(p^+|c)$ is thus estimated based on the product of two frequency statistics: the *premise frequency* $pf(p^+, c)$, i.e., the frequency of using premises similar to p to support claims in c's cluster, and the *inverse claim frequency* $icf(p^+)$, which describes the inverse number of claim clusters for which p is used as support. Let $\gamma : C \to \Gamma$ be a function that assigns to a claim $c_i \in C$ its corresponding claim cluster γ_k and likewise $\pi : \mathcal{P} \to \Pi$ a function that assigns to a premise $p_i \in \mathcal{P}$ its corresponding premise cluster π_j. Then, the two components are formalized as follows:

i) $pf(p^+, c) = |\{p'^+ \to c' : p' \in \pi(p^+), c' \in \gamma(c)\}|$

ii) $icf(p^+) = \log\left(\frac{|\Gamma|}{|\{\gamma \in \Gamma:\ \exists p'^+ \in \pi(p^+), \exists c' \in \gamma \text{ such that } p'^+ \to c'\}|}\right)$

Using this, $P(p^+|c)$ can be estimated with

$$P_{pf\text{-}icf}(p^+|c) = \frac{pf(p^+, c) \cdot icf(p^+)}{Z} \tag{4}$$

where Z is a normalization constant, since the outcome is not necessarily in the interval $[0, 1]$.

4 QUALITY-AWARE ARGUMENT RETRIEVAL

An obvious drawback of the probabilistic framework of our previous work [10] is that it works exclusively with frequencies, but does not take argument quality into account. The different dimensions of argument quality could have different effects on people, such as logic traceability, which may be contrary to emotive effects. Although frequency could also be considered as a quality dimension, we reserve this term to describe only those quality dimensions that can have varying effects on different users. In this section we will extend the framework to incorporate argument quality into the ranking. After adding argument quality to the framework, we discuss potential estimators for it.

4.1 Quality-Aware Probabilistic Framework

Let $\mathcal{D} = \{d_1, d_2, \ldots\}$ be the set of all argument quality dimensions, for example those introduced by Wachsmuth et al. [37]. In a practical ranking task with a real user, not all of these quality dimensions may be of equal importance, and some may not be important at

all. We thus consider the subset $\Delta \subseteq \mathcal{D}$ of quality dimensions of interest to the user that should be considered for ranking.

Our quality-aware ranking framework now extends the probabilistic ranking framework as follows: instead of considering the probability $P(p^+|c)$ that a user selects premise p from claim c among all its supporting premises (see Section 3.1), we make explicit that the user has certain quality dimensions in mind. This leads to the conditional probability $P(p^+|c, \Delta)$, i.e., the probability that the user picks premise p^+ from c among all its supporting premises, preferring premises that are of high quality in all argument quality dimensions $d \in \Delta$ that are of interest to the user. Note that in reality, users will usually not make an explicit choice of their preferred quality dimensions, so identifying the subset of quality dimensions to use here is part of the problem. We can now extend Equation 2 and obtain the quality-aware premise probability:

$$P(p^+|q, \Delta) = \sum_{c \in C: p \to c} P(c|q) \cdot P(p^+|c, \Delta) \tag{5}$$

Note that it is not necessary to include argument quality in the first component of this equation, $P(c|q)$, since a claim is not argumentative on its own [14].

The probability $P(\pi_j^+|q)$ for a user to pick a premise cluster π_j^+ (Equation 3) naturally expands to its quality-aware variant:

$$P(\pi_j^+|q, \Delta) = \sum_{p^+ \in \pi_j^+} P(p^+|q, \Delta) \tag{6}$$

4.2 Estimators for the Quality-Aware Premise Probability

A core building block for estimating the quality-aware premise probability is estimating the quality of the premise with respect to a quality dimension. In the literature, two approaches for this estimation have been published: (1) directly estimating the quality score of a premise, for example proposed by Wachsmuth et al. [37] or Gleize et al. [14], and (2) estimating the relative order of two premises with respect to a claim, i.e., which of two premises is 'better', for example proposed by Habernal and Gurevych [15].

As already stated in Section 2, the latter method not only allows us to use more training data, it also has the advantage that a ranking can be easily extended for unseen arguments, as a new argument can always have a better rating than the current best one which might already have the highest possible score. Thus, we use this approach and train a classifier for each quality dimension d that, given two premises p_1 and p_2 and a claim c, predicts if p_1 is better than p_2 with respect to c and d, or short $p_1 >_d^c p_2$. We now derive a ranking of all premises of a claim c with the same stance, with respect to a single quality dimension d, by counting, for each premise, how often it was estimated to be better than other premises. We denote this count as the *dimension convincing frequency* $dcf(p^+, c, d)$ which is defined as follows:

$$dcf(p, c, d) = |\{p >_d^c p' : p' \to c \wedge p \uparrow\uparrow p'\}| \tag{7}$$

Here, $p \uparrow\uparrow p'$ denotes that p and p' are premises with the same stance towards c. Note that we omit c for notational simplicity since it is clear from the context. Ordering the premises of c by descending $dcf(p, c, d)$ now yields a ranking of premises for claim c in descending estimated quality for dimension d. This allows us to directly estimate the quality-aware premise probability $P_{dcf}(p^+|c, d)$ for a single dimension d as

$$P_{dcf}(p^+|c, d) = \frac{1 + dcf(p^+, c, d)}{Z} \tag{8}$$

where $Z = \sum_{p^+ \to c}(1 + dcf(p^+, c, d))$ is a normalization constant. Note that we use Laplace Smoothing [22] to avoid probabilities with value 0.

To extend this towards multiple dimensions, we can simply multiply the per-dimension probabilities, which leads to the following equation:

$$P_{dcf}(p^+|c, \Delta) = \prod_{d \in \Delta} P_{dcf}(p^+|c, d) \tag{9}$$

We could now directly use P_{dcf} for the premise probability in Equation 5. However, the original idea of using premise frequencies has also performed reasonably well in the experiments of our prior work [10], so it may be worthwhile considering combinations of the two. We therefore will evaluate the following actual premise probabilities in our experimental evaluation:

a) the plain quality-aware premise probability $P_{dcf}(p^+|c, \Delta)$ (*plain family*)

b) the product of the quality-aware premise probability and the frequency-based premise probability (*product family*), i.e., $P_{pflcf \cdot dcf}(p^+|c, \Delta) = P_{pflcf}(p^+|c) \cdot P_{dcf}(p^+|c, \Delta)$

c) the average of the quality-aware premise probability and the frequency-based premise probability (*average family*), i.e.,

$$P_{avg}(p^+|c, \Delta) = \frac{P_{pflcf}(p^+|c) + P_{dcf}(p^+|c, \Delta)}{2}$$

5 IMPLEMENTATION AND EVALUATION

In this section we describe the concrete implementation of our proposed quality-aware ranking framework and present results of an experimental evaluation. For this purpose we used the dataset from Ajjour et al. [1], which is also used in Args [36], and set up our argument retrieval system. We evaluate the result quality with 50 queries provided by the CLEF lab Touché [4]. To estimate the quality of an argument with respect to the different dimensions we trained classifiers on a different dataset by Wachsmuth et al. [37] that provides data labeled according to several quality dimensions. Figure 1 visualizes the two-step retrieval system.

5.1 Classifier for Predicting Argument Quality

First, we address the development of a classifier for ranking arguments regarding specific quality dimensions. For this purpose, we consider the dataset Dagstuhl-15512 ArgQuality Corpus by Wachsmuth et al. [37] that provides arguments labeled according to several argument quality dimensions. For 32 (issue,stance) pairs with ten premises each, three experts have annotated the quality of 15 different dimensions on a scale from 1 (low) to 3 (high). An example for an issue is "*is the school uniform a good or bad idea*" and the stances for this issue are "*good*" or "*bad*". In the following, we will use the term *query* for (issue,stance) pairs synonymously. For these 320 arguments, we focus on the three main dimensions *cogency*,

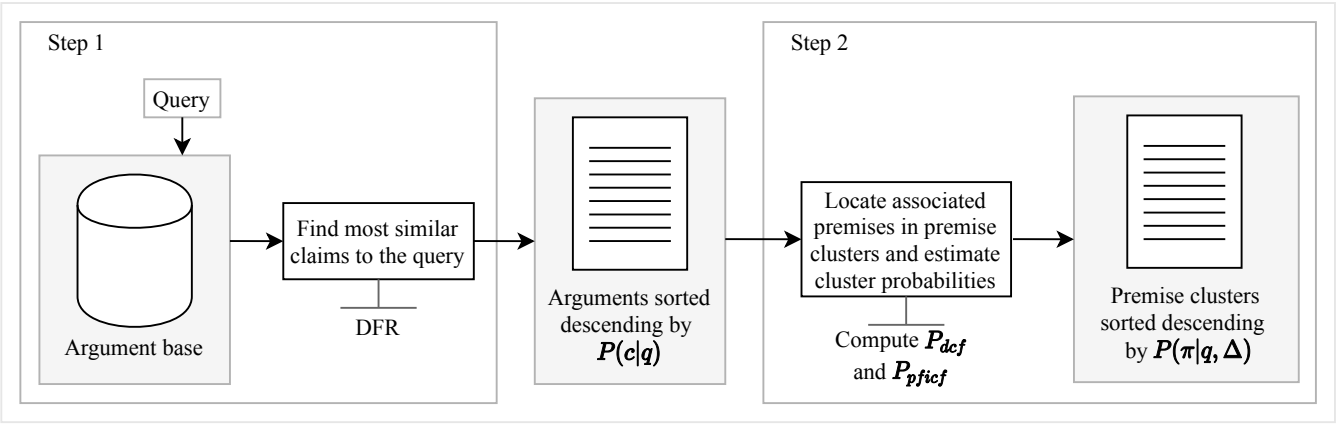

Figure 1: Visualization of the scheme for the two-step retrieval.

reasonableness, and *effectiveness* (see Section 2) in the evaluation and train classifiers for them.

However, since 320 arguments are most likely not sufficient for learning, we derive from this dataset another one with larger volumes. We follow the approach of Habernal and Gurevych [15] and do not consider the final ratings of the annotators, but derive for each of the 32 (issue,stance) queries new pairs in the form of (premise_1,premise_2), considering all combinations of premises of that query. For each of the three dimensions, we generate a new dataset where each pair is automatically labeled with "1" or "2" based on the original assessments. Here, the label "1" means that premise_1 is better with respect to a specific dimension and analogously "2" if premise_2 is better. We infer this label from the mean values of the three annotators' ratings per dimension and premise. Pairs with equal scores were removed from the datasets.

This resulted in 2,046 pairs for cogency, 2,074 for reasonableness, and 1,970 for effectiveness, that were used to train a classifier for each dimension that predicts, given two premises and a query, which premise is better for this query with respect to this dimension. Before training, all entries in the dataset were shuffled to avoid learning the structure. These classifiers can then be used to compute $dcf(p, c, d)$ as described in Section 4.2.

To represent both the premises and the query, their embeddings were first calculated. We used Sentence-BERT (SBERT) [25], which works on a siamese and triple network, since it claims to produce better embeddings than BERT [9]; it represents an input string with a vector of size 1,024. We used the model "`roberta-large-nli-stsb-mean-tokens`" because it provides the best performance for the semantic textual similarity task (STS-task).[6] Given the three embeddings (of the two premises and the query where the latter consists of the concatenated strings of the corresponding issue and stance), we then calculate the sum, the difference, and the product of each dimension of the premises for the query pointwise before concatenating the two vectors to obtain a new vector of length 6,144, which is the input to the classifier. Since we have 32 (issue,stance) pairs, we evaluated the process with leave-one-out cross-validation, i.e., we trained with 31 (issue,stance) pairs and their premises and tested with the remaining (issue,stance) pair and its premises. We measured the performance of seven standard classifiers. Those are Logistic Regression [23], Random Forest [20], Naive Bayes, Support Vector Machine [8], Gradient Boosting [5], K Nearest Neighbours [12], and Stochastic Gradient Descent [13] with standard parameters, that are all provided by Scikit-Learn [24].

Table 1 shows the results of the classifiers for each dimension. All classifiers yielded comparatively high accuracy values, most likely they received a boost because we generated all possible combinations and thus also included symmetrical pairs. Note that there is a trade-off which classifier produces better results for learning with additional symmetric pairs or learning with too little data. Of these, the Random Forest classifier delivered the best performance for *Cogency* and *Effectiveness*. Though the difference to Random Forest was very small, Logistic Regression achieved the best performance for *Reasonableness*.

In order to incorporate correction for multiple tests, we performed Tukey's HSD (honestly significant difference) test [29] with $p = .05$ on the accuracy values of the 32 folds showing no significant differences between Logistic Regression and Random Forest for the three dimensions (except Stochastic Gradient Descent). But there were significant differences between the two classifiers for the other dimensions (except Logistic Regression and Gradient Boosting for cogency). Shapiro-Wilk tests [31] delivered that the sets are normally distributed with exceptions of Naive Bayes for effectiveness, which is a negligible problem since Tukey's HSD tests are relatively robust to violations of the normal distribution assumption [19].

5.2 Dataset and Implementation

We evaluate our quality-aware argument retrieval system based on a dataset proposed by Ajjour et al. [1] that consists of 387,606 arguments in the form of (claim,premise) pairs. This data was extracted from several debate portals such as debate.org or idebate.org. It is also the dataset on which Args runs. Furthermore, it is the official dataset of the CLEF lab Touché which goal is to find the best arguments to 50 designated queries. By applying different datasets for training and evaluation, our findings get higher impact.

[6] https://github.com/UKPLab/sentence-transformers

Table 1: The accuracy values achieved by the classifiers for predicting the three main quality dimensions of arguments calculated with 32-fold-cross-validation. LR = 'Logistic Regression', RF = 'Random Forest', NB = 'Naive Bayes', SVM = 'Support Vector Machine', GB = 'Gradient Boosting', KNN = 'K Nearest Neighbours', SGD = 'Stochastic Gradient Descent'. All values are rounded to three decimal places. The highest values are shown in bold, the lowest are underlined.

	Accuracy		
Classifier	**Cogency**	**Reasonableness**	**Effectiveness**
RF	**.971**	.972	**.977**
LR	.958	**.976**	.97
SGD	.951	.964	.965
GB	.932	.942	.952
SVM	.918	.917	.922
KNN	.887	.89	.902
NB	<u>.792</u>	<u>.784</u>	<u>.778</u>

The implementation of our extended framework mostly follows the implementation proposed in our previous work [10]: as the arguments in the ARGS corpus consist of exactly one claim and one premise, we first grouped all premises that have the textually same claim and thus derived more complex arguments, i.e., 72,125 claims with an average of 5.37 and an median of 5 premises per claim. The new obtained arguments now have one claim with premises sizing from 1 to 2,539. A claim with smallest of the associated premises is "*Soler power energy production varies with the seasons.*", and with the largest is "*Abortion*".

To compute the clusters, we first calculated the embeddings of the claims and premises using SBERT [25] (with the same settings as for the classifiers) as opposed to our prior work where we used BERT [9]. Similar to this approach, we calculated the clusters with agglomerative clustering [16] using Euclidean distance and the average linkage method [32] by applying the scripting language R and the packages STATS and FASTCLUSTER. This resulted in 13,031 claim clusters and 70,314 premise clusters. The average, median, minimum, and maximum values of claim and premise clusters are 5.53, 5, 2, and 33 for claims and 5.51, 5, 1, and 1,867 for premises, respectively. Examples for a claim within a claim cluster with the smallest size is "*dogs*", and with largest size sentences often starting with "*There are no such thing as . . .* ". Analogously, a premise cluster with the smallest size (1) is a nonsense premise that starts with "*Agenda 21 is a bad decision.*" followed by hundreds of dots, while a premise from the cluster with the largest size is "*I accept*".[7]

We then indexed the claims and the premises with APACHE LUCENE (version 8.4.1), which also stores the corresponding claim cluster id or premise cluster id, respectively. Note, that in our previous work [10] we approximated the clustering of premises by clustering them ad-hoc at query time. While in this work we preclustered also the premises, the prior work clustered a subset determined by adding the ten most similar premises to the premises contained in the claims in the claim clusters, using the textual similarity method BM25. However, since preclustering of nearly 400k points with 1,024 dimensions each requires a lot of time, we first clustered the set with k-means [21] with k=4 and then conducted an agglomerative clustering on these subsets.[8] Similar to our prior work we choose the longest premise from a cluster as representative.

The first step of evaluating a query is to identify the most similar result claims, for which we use DFR [2]. Then all claims that are in the same clusters are located as well as their directly associated premises and the premises with the same cluster ids, which is the final set of candidate premises. For each candidate premise, we calculate the probabilities shown in Equation 6. Since $dcf(p, c, d)$ (see Section 4.2) calculates the quality of a premise to the claim and not to the query, we could also precalculate and store these frequencies for all dimensions. Measuring the quality of a premise to the claim and not the query is no drawback here, as this is compensated by the component $P(c|q)$ in Equation 5.

5.3 Evaluation Setup

We now discuss the setup for the evaluation of our approach. As baseline we implemented the original framework of our prior work [10]. However, while this work used the classic BERT implementation, we now use SBERT for both the baseline such that we can measure the impact of taking argument quality into account on result quality.

Next we evaluate the rankings calculated based on the final cluster probabilities of Equation 6 with the three different estimators from Section 4.2 for the three main argument quality dimensions $d \in \Delta$ with Δ = {cogency, reasonableness, effectiveness}, both for each dimension separately as well as all combined, i.e., Δ. Hence, together with the baseline we compare $1 + 3 \cdot 4 = 13$ methods.

For the evaluation, we used 50 widely spread queries provided by the CLEF lab Touché and evaluate them on the ARGS corpus [1]. Examples for such queries are "*Are Social Networking Sites Good for Our Society?*" or "*Is Homework Beneficial?*". However, Touché has run for the first time in 2020 and has not yet produced assessments for these queries. We therefore pooled, for each query, the ten best premises from each method, i.e., representatives of premise clusters. This resulted in overall 1,376 premises for the 50 queries.

Then, two annotators (one expert in computational argumentation and one expert from political science) assessed each premise with respect to the three dimensions' qualities on the scale from 1 (low) to 3 (high) by following the guidelines of Wachsmuth et al. [37] who used the same scale when they annotated the argument quality dimensions. Similar to them, we also added a "cannot judge" option. The relevance of the premise with respect to the query was also implicitly evaluated, and non-relevant premises were assigned a valued of 1. Note that only the query and the premise were shown to our annotators. Any other information, such as the actual result claim to which the premise belongs, was hidden from the assessors in order to avoid biased decisions. We implemented a dedicated tool for the assessors they used for their annotations; Figure 2 shows an example screenshot of the tool.

Altogether, the annotations with three dimensions for the 1,376 premises per assessor took about 65 hours each (approximately 21 premises per hour). Depending on the difficulty and length of the

[7] People in debate portals very often state that they accept to debate towards a specific topic, leading to the huge size of the premise cluster.

[8] For computing k-means we used the R package KNOR and used 25 starting points.

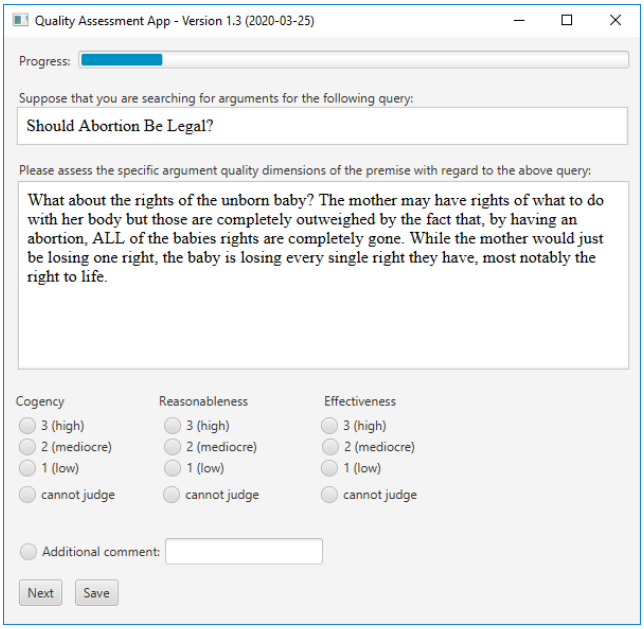

Figure 2: Screenshot of the quality assessment app the annotators used to decide the quality of arguments for three dimensions.

premises, the annotators assessed between two and four queries (the premises per query range from 21 to 34; 27.5 on average) per day in order to be able to perform the task with the highest possible concentration. To reduce learning effects as far as possible, we shuffled the ordering of the premises to the query before the assessments. Table 2 shows the distribution of the $2 \cdot 3 \cdot 1,376 = 8,256$ assessments made.

Table 2: The distribution of the assessment scores of the two annotators.

Argument Quality Dimension	Score 3	Score 2	Score 1	Cannot judge
Cogency	868	601	1,003	280
Reasonableness	569	1,059	846	278
Effectiveness	625	718	1,131	278

The inter-annotator agreement calculated with Krippendorff's α [18] for cogency, reasonableness, and effectiveness on a nominal scale and on an interval scale are shown in Table 3. Although the table indicates the difficulties in the annotation process, the agreements range from slight to substantial for the interval metric. The highest level of disagreement is obviously in the nominal metric of the dimension reasonableness. This dimension is particularly difficult as here the annotators have to take into account the public opinion. While one annotator more often assigned the value 1, the other more often assigned the value 2.

Table 3: The inter-annotator agreement values for the three argument quality dimensions measured with Krippendorff's α. All values are rounded to three decimal places.

Argument Quality Dimension	Nominal metric	Interval metric
Cogency	.315	.514
Reasonableness	.013	.307
Effectiveness	.389	.629

5.4 Evaluation Results and Error Analysis

The quality of the computed results is measured in terms of the nDCG metric [17] for each dimension. Here, the (undiscounted) gain of each result corresponds to the mean assessment scores minus 1, yielding a final score $x \in [0, 2]$ per premise and dimension; if a premise was assessed as "cannot judge", its gain is set to 0. Unassessed results would be assigned a gain of 0, but since we consider only cutoffs up to 10, all retrieved results are assessed. We measure nDCG at cutoffs 1,5,10, but we report only results for cutoffs 1 and 10 for space reasons; results for cutoff 5 were very similar to the results for cutoff 10. In a first analysis, we will take premises that could not be judged into account (henceforth called *noisy dataset*). However, since these premises often correspond to useless statements such as "*I accept*" that would not be included in high-quality argument collections, we will ignore these premises in a second analysis (henceforth called *curated dateset*). Further, we will show results of statistical significance tests of the two datasets.

5.4.1 Analyzing the Noisy Dataset. Table 4 shows the results for the first analysis. The table indicates how well a user would be satisfied if she valued a certain quality dimension, and how well a user would be satisfied if she expected good premises regarding all the three quality dimensions together. As we can infer from the table, the methods using the premise probabilities from the plain family, i.e., P_{dcf} that rely exclusively on the classifiers perform best by far for all cutoffs. The baseline P_{pflcf} which relies solely on the frequencies performs poorly. The methods from the product family, i.e., $P_{pflcf \cdot dcf}$ are better than the baseline, but still perform worse than those from the plain family.

Significance tests on the 50 accuracy values per method and dimension were made for cutoff value 1, 5, and 10. A Shapiro-Wilk test [31] delivered that the quantities are not always normally distributed. Since the strict view that the normal distribution is a prerequisite for Tukey's HSD test is outdated [19], we carried it out. Regarding the cutoff value 1, methods from the plain family do not only significantly perform better than the baseline, they also outperform the families product and average for the three dimensions plus the combination with $p < .002$. Methods from the product family are in almost all cases significantly better than the baseline with $p < .008$ and in most cases also significantly better than the methods from the average family with $p < .024$. The observation for the cutoff value 10 shows that all methods from the plain family are significantly better than all other methods for all dimensions plus their combination with $p < .036$. Furthermore, there are significant differences over the baseline for P_{avg} ($p < .017$)

Table 4: Noisy dataset. nDCG values of the baseline compared to the twelve methods for the three estimators. CO:= cogency, RE:= reasonableness, EF:= effectiveness, and the sum of the estimators. All values are rounded to three decimal places. The highest values are shown in bold, the lowest are underlined.

premise probability	Δ	cogency		reasonableness		effectiveness		{CO, RE, EF}	
		nDCG@1	nDCG@10	nDCG@1	nDCG@10	nDCG@1	nDCG@10	nDCG@1	nDCG@10
P_{pflcf}	-	.010	.258	.022	.297	.010	.245	.014	.270
P_{dcf}	{CO}	.515	.630	.585	.680	.462	.565	.521	.634
P_{dcf}	{RE}	**.640**	.642	**.640**	.684	**.567**	.574	**.617**	.642
P_{dcf}	{EF}	.560	.638	.612	.678	.505	.579	.562	.640
P_{dcf}	{CO, RE, EF}	.555	**.647**	.588	**.702**	.497	**.589**	.549	**.654**
$P_{pflcf \cdot dcf}$	{CO}	.180	.407	.207	.458	.158	.363	.183	.416
$P_{pflcf \cdot dcf}$	{RE}	.250	.475	.292	.532	.192	.425	.243	.484
$P_{pflcf \cdot dcf}$	{EF}	.120	.390	.150	.427	.112	.335	.127	.390
$P_{pflcf \cdot dcf}$	{CO, RE, EF}	.080	.356	.087	.395	.070	.321	.079	.362
P_{avg}	{CO}	.010	.390	.037	.422	.02	.348	.022	.392
P_{avg}	{RE}	.075	.410	.077	.443	.065	.370	.071	.414
P_{avg}	{EF}	.030	.384	.037	.417	.030	.349	.032	.389
P_{avg}	{CO, RE, EF}	.030	.368	.037	.409	.03	.340	.032	.377

and $P_{pflcf \cdot dcf}$ ($p < .001$) for reasonableness and $P_{pflcf \cdot dcf}$ ($p < .022$) for cogency.

We examined the rankings of the premises of the baseline P_{pflcf} as well as P_{dcf} with $\Delta = \{CO, RE, EF\}$ to find out why P_{dcf} performs well but P_{pflcf} comparatively poorly. Since we work with an uncurated dataset here and *pflcf* uses only frequencies of claims and premises, the top 1 results of the baseline very often contained statements such as "*Extend*", "*I accept*", or "*First round is for acceptance*", etc., because some debate portals often let discussions take place over several rounds and some users first tell the community that they accept the debate. In contrast, the methods from the plain family utilizing the classifiers here recognized that there is no argumentative content and therefore rank them low. This clearly shows the need for taking argument quality into account in the retrieval process. Obviously, at least one of the human annotators assessed them with "cannot judge", which were considered equal to the relevance scores as assessment "low" in the first analysis and led to lower nDCG values for P_{pflcf} in Table 4.

5.4.2 Analyzing the Curated Dataset. In order to exclude the impact of these premises, we performed another evaluation where the pairs evaluated with "cannot judge" by at least one annotator were ignored. Table 5 shows the results of this experiment. Contrary to the noisy dataset, the baseline P_{pflcf} now performs much better. We see that the procedures from the plain family still perform slightly better for cutoff value 10, but no longer for cutoff 1.

In contrast to the noisy dataset, there are no significant differences (Tukey's HSD test with $p = .05$) between baseline and all methods for cutoff value 1 for both the three dimensions and the combination. For the cutoff value 10, the methods from the plain family are significantly better than the baseline for $d \in \Delta \backslash \{EF\}$ with $p < .048$. Furthermore, all methods from the average family are significantly better than the baseline for RE with $p < .038$.

A second manual random sampling of the top 1 results of baseline and the aforementioned method P_{dcf} with $\Delta = \{CO, RE, EF\}$ provided the insight that the latter simply provides more convincing premises than the former, indicating that the classifiers are robust.

5.5 Evaluation with another Dataset

In order to strengthen the quality of our findings so far, we also applied our methods to the dataset of our prior work [10], which contains 63,250 claims and approximately 695k premises that were extracted from four debate portals. The dataset provides 30 query claims on the topic "energy" together with manually formed premise clusters assessed with regard to their relevance on a three-fold scale as "very relevant", "relevant", or "not relevant". The quality of arguments was not taken into account during the assessment, only relevance of the premises with respect to the query. In the following, this dataset is referred to as dataset$_{energy}$. Following the approach described in that paper, we evaluated two tasks: In Task B the selection of a representative from a cluster is evaluated using a simplified variant of α-nDCG (with $\alpha = 1$), in Task A a list of premises by generating all possible result lists by including all combinations of the premise clusters and comparing it with the mean average nDCG values. Table 6 shows the evaluation of Task A and Task B. In contrast to the previous section's dataset (in the following dataset$_{spread}$) it is remarkable that (1) the differences of the examined methods (especially to the baseline) are not as great here as in dataset$_{energy}$, (2) the nDCG values are higher, and (3) the baseline performs relatively well. Shapiro-Wilk tests revealed that the accuracy values for the 30 queries at cutoff 5 and 10 for Task A and Task B do not vary from a normal distribution. Subsequent Tukey's HSD tests to determine whether there are significant differences between the methods used for baseline or different estimators with respect to a quality dimension revealed that there were no significant differences for neither Task A nor Task B for both cutoff values. We justify these differences as follows: The topic of energy is more likely to be discussed by people with specialist knowledge than the topics discussed in dataset$_{spread}$, therefore it is more difficult to find differences in the premises' structures. The higher nDCG values

Table 5: Curated dataset. nDCG values of the baseline compared to the twelve methods for the three estimators and for the sum of them. CO:= cogency, RE:= reasonableness, EF:= effectiveness. All values are rounded to three decimal places. The highest values are shown in bold, the lowest are underlined.

premise probability	Δ	cogency		reasonableness		effectiveness		{CO, RE, EF}	
		nDCG@1	nDCG@10	nDCG@1	nDCG@10	nDCG@1	nDCG@10	nDCG@1	nDCG@10
P_{pflcf}	-	.545	.504	.608	.575	.518	.478	.56	.526
P_{dcf}	{CO}	.535	.644	.605	.696	.482	.578	.541	.649
P_{dcf}	{RE}	**.645**	.656	.66	.697	.567	.584	.625	.655
P_{dcf}	{EF}	.60	.656	.652	.699	.545	.597	.602	.659
P_{dcf}	{CO, RE, EF}	.61	**.668**	.647	**.725**	.532	**.609**	.598	**.676**
$P_{pflcf \cdot dcf}$	{CO}	.515	.568	.572	.637	.467	.511	.519	.58
$P_{pflcf \cdot dcf}$	{RE}	.52	.578	.63	.645	.46	.518	.536	.588
$P_{pflcf \cdot dcf}$	{EF}	.51	.564	.597	.618	.437	.494	.516	.567
$P_{pflcf \cdot dcf}$	{CO, RE, EF}	.565	.568	.622	.634	.52	.513	.57	.579
P_{avg}	{CO}	.56	.627	.623	.685	.507	.564	.565	.634
P_{avg}	{RE}	.595	.636	.65	.691	.522	.576	.589	.643
P_{avg}	{EF}	.545	.636	.61	.686	.495	.581	.553	.642
P_{avg}	{CO, RE, EF}	.625	.635	**.688**	.693	**.585**	.583	**.635**	.644

probably result from the fact that for the dataset$_{energy}$ claims were used as queries which also originate from dataset$_{energy}$, whereas for dataset$_{spread}$ queries were used which never match a claim 1 on 1, so the retrieval task considered now is easier. Furthermore, some queries used for dataset$_{spread}$ simply have no or very few matching results since we work with previously mined arguments. Most probably the main reason for the better performance of the methods in dataset$_{spread}$, which include argument quality dimensions, can be found in the topics. While dataset$_{energy}$ only contains topics from one topic where most premises will argue based on facts, dataset$_{spread}$ is wide spread with 50 queries on a large number of topics. Especially topics concerning "*abortion*" or "*gun laws*" are much more emotionally loaded and are discussed more passionately, so the classifiers are more suitable there. To conclude, one can say that P_{pflcf} performs very well when the dataset is curated. Including argument quality dimensions can further improve the premise retrieval, especially in noisy datasets.

6 CONCLUSION AND FUTURE WORK

In this paper we propose a framework for quality-aware argument retrieval, building on and extending a probabilistic ranking framework with argument quality dimensions. Using classifiers for the three main dimensions cogency, reasonableness, and effectiveness, our experimental evaluation showed that argument quality is essential for achieving good retrieval quality, especially on noisy datasets. We will provide the argumentation community with both the 1,376 premises labeled for three argument quality dimensions and an API (via Docker and Web interface).

Here, we considered premises from debate portals, which are partially from moderated websites and are real-world opinions. However, clustering and ranking premises remains very difficult, even for human annotators because sometimes posts are very long, address several aspects, and contain non-argumentative text. Future work needs to extract arguments in the sense of argumentation theory from posts, i.e., on the one hand to detect argumentative and non-argumentative text spans, such as "*I will present my arguments*" or "*I accept the debate*" and on the other hand to separate arguments by their meaning. Sometimes definitions of terms are introduced before the actual arguments, followed by arguments. It might also be necessary to consider rephrasing and inclusion of knowledge graphs. Starting from the labeled dataset, it is now possible to train a new classifier, which is specially designed for this corpus.

Table 6: Evaluations of Tasks A and B: List of premise clusters and the choice of representatives, respectively. Average nDCG values (Task B) and mean average nDCG values (Task A) of the baseline *pf · icf* compared to the twelve methods for the relevance from dataset$_{energy}$. CO:= cogency, RE:= reasonableness, EF:= effectiveness. All values are rounded to three decimal places. The highest values are shown in bold, the lowest are underlined.

		Task B		Task A	
premise probability	Δ	average nDCG@5	average nDCG@10	mean average nDCG@5	mean average nDCG@10
P_{pflcf}	-	.72	.677	.67	**.678**
P_{dcf}	{CO}	.695	.667	.646	.649
P_{dcf}	{RE}	.721	.67	.646	.65
P_{dcf}	{EF}	**.724**	.681	**.679**	.668
P_{dcf}	{CO, RE, EF}	.699	.659	.645	.648
$P_{pflcf \cdot dcf}$	{CO}	.721	**.689**	.668	.66
$P_{pflcf \cdot dcf}$	{RE}	.716	.672	.646	.648
$P_{pflcf \cdot dcf}$	{EF}	.715	.68	.663	.662
$P_{pflcf \cdot dcf}$	{CO, RE, EF}	.708	.678	.661	.658
P_{avg}	{CO}	.688	.66	.64	.641
P_{avg}	{RE}	.701	.659	.638	.638
P_{avg}	{EF}	.718	.672	.666	.656
P_{avg}	{CO, RE, EF}	.701	.661	.644	.649

ACKNOWLEDGMENTS

We would like to thank Tobias Wiesenfeldt for his invaluable help in annotating the premises.

This work has been funded by the Deutsche Forschungsgemeinschaft (DFG) within the project ReCAP, Grant Number 375342983 - 2018-2020, as part of the Priority Program "Robust Argumentation Machines (RATIO)" (SPP-1999).

REFERENCES

[1] Yamen Ajjour, Henning Wachsmuth, Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. 2019. Data Acquisition for Argument Search: The args.me Corpus. In *KI 2019: Advances in Artificial Intelligence - 42nd German Conference on AI, Kassel, Germany, September 23-26, 2019, Proceedings (Lecture Notes in Computer Science)*, Vol. 11793. Springer, 48–59. https://doi.org/10.1007/978-3-030-30179-8_4

[2] Gianni Amati and C. J. van Rijsbergen. 2002. Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems* 20, 4 (2002), 357–389. https://doi.org/10.1145/582415.582416

[3] J. Anthony Blair. 2012. *Groundwork in the Theory of Argumentation.* Argumentation Library, Vol. 21. Springer Netherlands. https://doi.org/10.1007/978-94-007-2363-4

[4] Alexander Bondarenko, Matthias Hagen, Martin Potthast, Henning Wachsmuth, Meriem Beloucif, Chris Biemann, Alexander Panchenko, and Benno Stein. 2020. Touché: First Shared Task on Argument Retrieval. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II (Lecture Notes in Computer Science)*, Vol. 12036. Springer, 517–523. https://doi.org/10.1007/978-3-030-45442-5_67

[5] Leo Breiman. 1997. *Arcing the edge.* Technical Report. Technical Report 486, Statistics Department, University of California at ....

[6] Elena Cabrio and Serena Villata. 2018. Five Years of Argument Mining: a Data-driven Analysis. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden.* 5427–5433. https://doi.org/10.24963/ijcai.2018/766

[7] Charles L. A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008.* ACM, 659–666. https://doi.org/10.1145/1390334.1390446

[8] Corinna Cortes and Vladimir Vapnik. 1995. Support-Vector Networks. *Mach. Learn.* 20, 3 (1995), 273–297. https://doi.org/10.1007/BF00994018

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers).* 4171–4186. https://aclweb.org/anthology/papers/N/N19/N19-1423/

[10] Lorik Dumani, Patrick J. Neumann, and Ralf Schenkel. 2020. A Framework for Argument Retrieval - Ranking Argument Clusters by Frequency and Specificity. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part I (Lecture Notes in Computer Science)*, Vol. 12035. Springer, 431–445. https://doi.org/10.1007/978-3-030-45439-5_29

[11] Lorik Dumani and Ralf Schenkel. 2019. A Systematic Comparison of Methods for Finding Good Premises for Claims. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019.* 957–960. https://doi.org/10.1145/3331184.3331282

[12] Evelyn Fix and J. L. Hodges Jr. 1952. Discriminatory analysis: Nonparametric discrimination: Consistency properties. *USAF School of Aviation Medicine, Project* (1952), 21–49.

[13] Jerome H Friedman. 2002. Stochastic gradient boosting. *Computational statistics & data analysis* 38, 4 (2002), 367–378.

[14] Martin Gleize, Eyal Shnarch, Leshem Choshen, Lena Dankin, Guy Moshkowich, Ranit Aharonov, and Noam Slonim. 2019. Are You Convinced? Choosing the More Convincing Evidence with a Siamese Network. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers.* 967–976. https://www.aclweb.org/anthology/P19-1093/

[15] Ivan Habernal and Iryna Gurevych. 2016. Which argument is more convincing? Analyzing and predicting convincingness of Web arguments using bidirectional LSTM. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers.* https://www.aclweb.org/anthology/P16-1150/

[16] Anil K. Jain and Richard C. Dubes. 1988. *Algorithms for Clustering Data.* Prentice-Hall.

[17] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems* 20, 4 (2002), 422–446. https://doi.org/10.1145/582415.582418

[18] Klaus Krippendorff. 1970. Estimating the Reliability, Systematic Error and Random Error of Interval Data.

[19] David M. Lane. 2018. All Pairwise Comparisons Among Means. Retrieved 05-August-2020 from http://onlinestatbook.com/2/tests_of_means/pairwise.html

[20] Andy Liaw and Matthew Wiener. 2002. Classification and Regression by randomForest. *R News* 2, 3 (2002), 18–22. http://CRAN.R-project.org/doc/Rnews/

[21] Stuart P. Lloyd. 1982. Least squares quantization in PCM. *IEEE Trans. Inf. Theory* 28, 2 (1982), 129–136. https://doi.org/10.1109/TIT.1982.1056489

[22] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval.* Cambridge University Press. https://doi.org/10.1017/CBO9780511809071

[23] Peter McCullagh and John A. Nelder. 1989. *Generalized Linear Models.* Springer. https://doi.org/10.1007/978-1-4899-3242-6

[24] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* 12 (2011), 2825–2830. http://dl.acm.org/citation.cfm?id=2078195

[25] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019.* Association for Computational Linguistics, 3980–3990. https://doi.org/10.18653/v1/D19-1410

[26] Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2019. Classification and Clustering of Arguments with Contextualized Word Embeddings. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers.* 567–578. https://www.aclweb.org/anthology/P19-1054/

[27] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at TREC-3. In *Proceedings of The Third Text REtrieval Conference, TREC 1994, Gaithersburg, Maryland, USA, November 2-4, 1994*, Vol. Special Publication 500-225. National Institute of Standards and Technology (NIST), 109–126. http://trec.nist.gov/pubs/trec3/papers/city.ps.gz

[28] Stephen E. Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval* 3, 4 (2009), 333–389. https://doi.org/10.1561/1500000019

[29] Tetsuya Sakai. 2018. *Laboratory Experiments in Information Retrieval - Sample Sizes, Effect Sizes, and Statistical Power.* The Information Retrieval Series, Vol. 40. Springer. https://doi.org/10.1007/978-981-13-1199-4

[30] Gerard Salton, A. Wong, and Chung-Shu Yang. 1975. A Vector Space Model for Automatic Indexing. *Commun. ACM* 18, 11 (1975), 613–620. https://doi.org/10.1145/361219.361220

[31] Samuel Sanford Shapiro and Martin B Wilk. 1965. An analysis of variance test for normality (complete samples). *Biometrika* 52, 3/4 (1965), 591–611.

[32] R. R. Sokal and C. D. Michener. 1958. A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin* 38 (1958), 1409–1438.

[33] Christian Stab, Johannes Daxenberger, Chris Stahlhut, Tristan Miller, Benjamin Schiller, Christopher Tauchmann, Steffen Eger, and Iryna Gurevych. 2018. ArgumenText: Searching for Arguments in Heterogeneous Sources. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 2-4, 2018, Demonstrations.* 21–25. https://www.aclweb.org/anthology/N18-5005/

[34] Christian Stab and Iryna Gurevych. 2014. Identifying Argumentative Discourse Structures in Persuasive Essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL.* 46–56. https://www.aclweb.org/anthology/D14-1006/

[35] Manfred Stede, Stergos D. Afantenos, Andreas Peldszus, Nicholas Asher, and Jérémy Perret. 2016. Parallel Discourse Annotations on a Corpus of Short Texts. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.* http://www.lrec-conf.org/proceedings/lrec2016/summaries/477.html

[36] Henning Wachsmuth, Martin Potthast, Khalid Al Khatib, Yamen Ajjour, Jana Puschmann, Jiani Qu, Jonas Dorsch, Viorel Morari, Janek Bevendorff, and Benno Stein. 2017. Building an Argument Search Engine for the Web. In *Proc. 4th Workshop on Argument Mining (ArgMining@EMNLP).* 49–59. https://doi.org/10.18653/v1/W17-5106

[37] Henning Wachsmuth, Benno Stein, Graeme Hirst, Vinodkumar Prabhakaran, Yonatan Bilu, Yufang Hou, Nona Naderi, and Tim Alberdingk Thijm. 2017. Computational Argumentation Quality Assessment in Natural Language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers.* 176–187. https://aclweb.org/anthology/E17-1017/