Ranking Passages for Argument Convincingness

Peter Potash, Adam Ferguson, Timothy J. Hazen
Microsoft Research Montreal
{pepotash,adfergus,tihazen}@microsoft.com

Abstract

In data ranking applications, pairwise annotation is often more consistent than cardinal annotation for learning ranking models. We examine this in a case study on ranking text passages for argument convincingness. Our task is to choose text passages that provide the highest-quality, most-convincing arguments for opposing sides of a topic. Using data from a deployed system within the Bing search engine, we construct a pairwise-labeled dataset for argument convincingness that is substantially more comprehensive in topical coverage compared to existing public resources. We detail the process of extracting topical passages for queries submitted to a search engine, creating annotated sets of passages aligned to different stances on a topic, and assessing argument convincingness of passages using pairwise annotation. Using a state-of-the-art convincingness model, we evaluate several methods for using pairwise-annotated data examples to train models for ranking passages. Our results show pairwise training outperforms training that regresses to a target score for each passage. Our results also show a simple 'win-rate' score is a better regression target than the previously proposed page-rank target. Lastly, addressing the need to filter noisy crowd-sourced annotations when constructing a dataset, we show that filtering for transitivity within pairwise annotations is more effective than filtering based on annotation confidence measures for individual examples.

1 Introduction

In online searches, results are typically presented to users ranked only by the relevancy of the results to the query. Search engines typically learn such relevancy through the positive reinforcement of user clicks. However, when queries address topics with multiple perspectives, some of which may be polarizing and divisive, search result click-through may reinforce biases of users contributing to the digital filter bubble or echo chamber phenomena (Barberá et al., 2015; Vaccari, 2013; Jamieson and Cappella, 2008; Wallsten, 2005).

To counter the filter bubble effect, search engines may seek to actively provide diverse results to topical queries (Yom-Tov et al., 2014), or even explicitly present arguments on different sides of an issue (Stab et al., 2018). In such scenarios, it is desirable to not only consider the relevancy of the diverse search results, but also their quality and convincingness. In our work, we seek to rank a collection of text passages by their argument convincingness, for use in Bing's multi-perspective search feature that presents arguments on different sides of a topical issue requested by a search query. An example of our use case and the goal of the model we aim to construct are presented in Table 1.

Habernal and Gurevych (2016) formally introduced the task of predicting argument convincingness to the language processing community by providing the first annotated corpus[1] (the UKP dataset), as well as providing initial experimental results on the dataset. The UKP dataset is annotated in a pairwise fashion: given two arguments with the same stance toward an issue, label which argument is more convincing. The implementation of pairwise annotation for this dataset is theoretically and practically grounded.

Motivated by the pioneering work of Thurstone (1927), pairwise labeling is a popular method for annotating items for attribute value (Heldsinger

[1] Although the ChangeMyView (CMV) (Tan et al., 2016) dataset had been published several months earlier, we believe the argumentation involved in the CMV dataset is more along the lines of debate and persuasion because commentators are trying to rebut the initial opinions and assertions made by the original poster. The same also holds for the dataset from Durmus and Cardie (2018).

Proceedings of the 6th Workshop on Argument Mining, pages 146–155
Florence, Italy, August 1, 2019.

Query: reasons why nafta is good	
Passages with a "Pro" stance	**Passages with a "Con" stance**
Candidate 1: NAFTA has six advantages. First, it quadrupled trade between Canada, Mexico, and the United States. That's because the agreement eliminated tariffs. Trade increased to $1.14 trillion in 2015. Second, it lowered prices. The United States imports Mexican oil for less than before the agreement.	**Candidate 1:** Is NAFTA a Bad Deal? The North American Free Trade Agreement (NAFTA) has come under fire recently, with some labeling it a disaster and claiming that it is the driving force behind the relocation of American firms like Ford Motor Company to Mexico.
Candidate 2: Because it helps in political interests. NAFTA is meant to lower tariffs and therefore create pro business alliances between the three signing nations. This allows for the U.S. to buy products cheaper from Canada and tears down the barriers to trade such as tariffs fees etc.	**Candidate 2:** Best Answer: see... the problem is... people who support NAFTA only compare it to either all out free trade... or no trade. trade is good and needed... but that doesn't mean it has to be, or should be FREE trade... so stop with these false comparisons of we have to trade...

Table 1: The table above shows the use-case for a ranking model for convincingness. Suppose a user has typed the query 'reasons why nafta is good'. Normally, this query will elicit links to texts that reflect only a positive stance toward the 'nafta' issue. Alternatively, a system can be designed to show arguments from *both* sides of the issue. In our system, we seek to select and present one passage to show for each side of the issue. Given passages that have been mapped to the pro and con sides of the issue, we will use our model to choose the best passage to show for each side of the issue. The above example illustrates a situation with two passage candidates for each of the pro/con sides, and our model needs to choose the most convincing one to display for each side.

and Humphry, 2010; Loewen et al., 2012). Recently, Shah et al. (2014) have conducted a suite of annotation experiments in order to empirically validate the belief that pairwise annotation is faster and more accurate than cardinal annotation for comparative tasks[2]. This paper presents a practical case study of a scenario where we have annotated data in a pairwise fashion and wish to train a model for ranking purposes.

The base model we use for predicting argument convincingness is an extension of the sum-of-embeddings model proposed by Potash et al. (2017). Our base model records state-of-the-art performance on the ranking subtask from the UKP dataset. Building on the base model, we explain two primary methods for going from pairwise data to a general ranking model: 1) Train a model that independently produces scores for each passage using a pairwise training paradigm to minimize a cross entropy objective function; 2) Assign real-valued scores to each passage, and train a model with a regression objective function to minimize the model's error against these scores. The second approach requires a method to pre-generate the real-valued passage scores used as the regression targets using only pairwise annotations. Towards this secondary goal, we test two approaches: 1) Following Habernal and Gurevych (2016), we generate PageRank (PR) (Page et al., 1999) scores using directed graphs derived from the labeled pairs; 2) We compute a simple 'Win-Rate' (WR) percentage based on how often a passage is rated more convincing against its competitor passages.

In order to test the robustness of the proposed techniques for using pairwise-labeled data to create a ranking model, we construct a new dataset for convincingness with a superior coverage of topics compared to the UKP dataset, which only has passages for 16 topics and roughly 1k total passages. In comparison, our dataset covers 3,234 topics, with roughly 30k total passages. The results of experiments on the large-scale dataset show that the best method for training a ranking model is to use the pairwise labels directly. Secondly, regarding the regression-based models, regressing to WR is better than PR, and even competitive with pairwise training. Finally, filtering data based on label confidence can actually hurt performance, although it can be beneficial to weight a pairwise model based on label confidence. Alternatively, removing query-passage sets where cycles appear

[2]In cardinal annotation, each individual example is assigned a score from a scale to signify the intensity of a given attributed being annotated.

in the directed graphs induced by the labels of passage pairs is a preferred method for data-filtering in our case study.

2 Related Work

In terms of predicting argument convincingness, only four authors have published results on the UKP dataset (Habernal and Gurevych, 2016; Chalaguine and Schulz, 2017; Potash et al., 2017; Simpson and Gurevych, 2018), with Potash et al. (2017) and Simpson and Gurevych (2018) posting state-of-the-art results on the pairwise classification[3] and ranking tasks, respectively. Simpson and Gurevych's model uses Gaussian Process Preference Learning (Chu and Ghahramani, 2005), which learns a mapping from input passage representations to real-valued scores.

Related to our use of label confidence to weight training examples, solving problems in NLP with models that leverage annotator agreement/confidence has previously been explored. Plank et al. (2014) and Alonso et al. (2015) use the information from individual annotations on examples to improve sequential (part-of-speech tagging) and structural (dependency parsing) tasks. Previously, Beigman and Klebanov (2009) had shown theoretically that noise from ambiguously-annotated examples are more harmful to certain learning models, namely the Voting Perceptron algorithm (Freund and Schapire, 1999).

Lastly, methods for ranking from pairs is a relevant research area for our work. Chen et al. (2013) adopt an active learning framework for the popular Bradley-Terry model (Bradley and Terry, 1952) in order to minimize the amount of annotations required to train a ranking model from pairwise data. Negahban et al. (2016) propose an algorithm, Rank Centrality, that works on a graph induced by pairwise annotations where node scores come from their stationary probability under a random walk. Chen and Suh (2015) improve upon Rank Centrality by introducing an algorithm that is specifically intended to recover the top k rankings via spectral initialization and continued refinement over the pairs with a maximum likelihood estimation.

[3]See Section 4.1 for more details of this model, as it is the basis for our approach for modeling argument convincingness. Moreover, the model from Simpson and Gurevych was not yet public as we were developing our model.

3 Dataset

Throughout the paper, we will refer to elements of our dataset using terms that form a hierarchy. At the top level, we use the term *topic*. A topic is an idea/issue devoid of a specific stance/assertion. Examples of topics are "coffee", "nafta", "margarine", and "fluoride". Within each topic are *queries*, which are search statements/questions that possess a specific thesis/stance with regard to its topic. For the topic "coffee", a query may be "is coffee good for you", which takes as the assertion: "coffee is good for you". An alternative query may be "is coffee bad for you". The third element of the hierarchy is a *passage*. A passage exists with respect to a query, and argues the position that is present in a query. Each query has multiple passages, all with the same stance toward a topic. In this section we describe the process of going from raw search data to a cleaned and annotated dataset with passages of the same topic and stance annotated for argument convincingness. The reason we want to have data annotated in this manner is it reflects the context in which we would plan to use the proposed model: we make the assumption that the input passages to be ranked are all on the same side of a stance related to a given issue, which, in a practical scenario, has been dealt with by upstream processing.

3.1 Dataset Creation

In order to test the utility of a convincingness model over a large variety of topics we created a dataset with larger topical coverage compared to the UKP data. We seeded the process with data collected for Bing's multi-perspective search feature, which was designed to show two short passages arguing for opposing stances of an issue expressed by a user query submitted to the system (e.g., "is coffee good for you"). The dataset consists of *topic*, *query*, *passage* triples. Each query conveys a *pro* or *con* sentiment for the expressed topic. Multiple potential passages are matched with each topic based on the Bing search engine's relevancy rankings with each passage assigned to the *pro* or *con* side of the topic based on a sentiment analysis classifier trained for the task. The passages themselves are snippets of text that have been scraped from the Web. For each query in a triplet, we have also automatically determined a paired query expressing the opposing stance (e.g., "is coffee bad for you") which we use to help

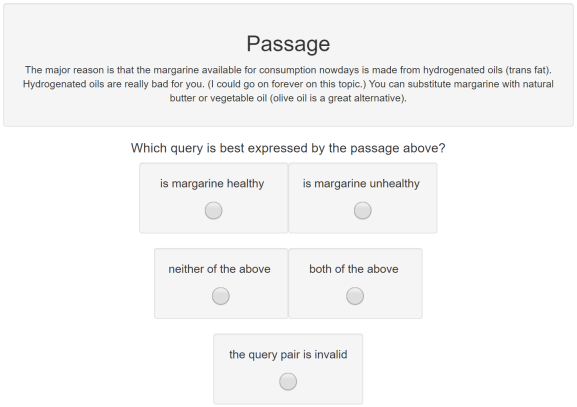

Figure 1: For stance annotation, workers are presented with a passage and a query pair, where each query is meant to reflect either a positive or negative stance toward an issue. The worker must choose which query best aligns with the passage.

validate the stance of passages as detailed below. The initial seed set contained 95,318 triples across 18,864 unique queries covering 3,439 topics. The initial annotations of the pro/con stances of queries and passages of the data available from the pre-existing system were created using automatic means (e.g., a sentiment analysis model) and were hence errorful. Additionally, no assessment of the convincingness of the passages had been conducted. Thus, we performed a two-stage manual annotation process on the dataset to (1) generate ground truth stance labels for query/passage pairs, and (2) generate pairwise convincingness assessments of passages associated with the same topic and stance.

3.2 Stance Annotation

Passage stance was determined by crowd workers judging which query from a positive-negative pair best aligns with a given passage. Workers also had the option of labeling that neither query aligns (i.e., the passage does not express a specific stance), or that both queries align with the passage (i.e., the passage provides arguments for both sides of the issue). To ensure that the query pairs themselves are valid, a fifth option specifying invalidity was provided for instances when a query is off-topic from the passage, is ambiguous in meaning, expresses multiple stances, or if both queries hold the same stance. Figure 1 shows the stance annotation layout. The goal of stance annotation is to identify pairs of passages that argue the same stance on a topic, as expressed by a query.

To contribute to the dataset, workers first had to read accompanying guidelines and examples then pass a qualification test with a grade of 70%. This test consisted of ten judgements made on passages pre-determined to represent two of each of the five available options. Feedback on the correct option was given after each judgement. If workers failed the initial qualifying set, they were provided with a second attempt on ten new instances to encourage learning and skill development.

Qualified workers who later hit an average speed less than six seconds per judgement[4], compared to the overall average of 16 seconds, or who had a low agreement score with other annotators, were removed from the task and their work was re-assigned to others. To prevent worker fatigue and ensure a wide breadth of participation, individual workers were prohibited from performing more than 10% of the available annotations tasks. The average number of annotations provided per worker was 1,178. Using this approach, each raw data point was annotated three times from a pool of 223 workers. The process yielded a total of 71,840 passage pairs associated with the same stance on the same topic.

3.3 Convincingness Annotation

Comparisons on passage convincingness are performed by workers judging which passage, from a pair with the same stance toward an issue, is more convincing. Refer to Figure 2 for the layout of the convincingness annotation. Workers are provided with tips on how to determine convincingness, such as evaluating topic deviation, use of facts, and citation of authority figures. To force workers to make a decision, workers were not given the option to rate the passages as equally convincing. Workers are instructed to consider the passage coherency and writing quality in the event of a tie in convincingness. Each of the 71,840 passage pairs identified during the stance annotation was annotated for convincingness by five different workers. We again applied techniques to pre-qualify workers and remove workers producing low-quality work.

Workers for this stage were also required to read guidelines and examples before passing a qualification test, though with an increased grade requirement of 80%. The test was composed of ten

[4]If a worker goes this speed, or faster, they are believed to be clicking answers randomly or *spamming*.

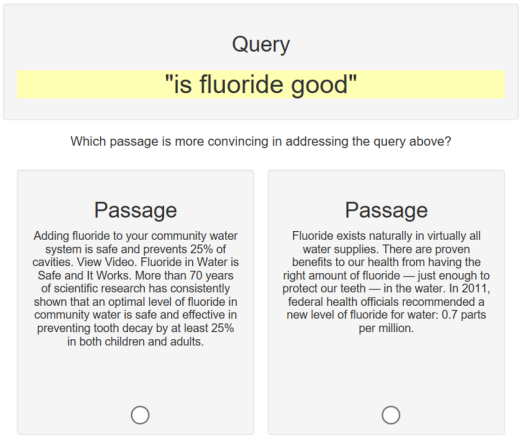

Figure 2: For convincingness annotation, workers are presented with two passages, with the same stance toward a topic (the search query), and are asked to label which passage is more convincing.

judgements, split evenly between easy and hard levels of difficulty, to be made on queries with passage convincingness predetermined. Feedback on the correct option was given after each judgement again and another attempt was provided in the event of failure, however this time without feedback. Workers whose average speed measured less than six seconds per judgement compared to the overall average of 20 seconds or who had low agreement scores with other annotators were blocked with their work being redone by the remaining annotators. The total number of judgements made per worker in this stage was limited to 5% of the total annotations with 12 reaching the limit and an overall average of 907 annotations per worker. A total of 71,840 query-passage pair sets were annotated five times each from a pool of 396 workers.

3.4 Constructing Passage Graphs

One key term for our work is **Passage Graph**, which is the result of using binary annotations of passage pairs to generate a directed graph. Mirroring the process from Habernal and Gurevych (2016), a directed graph is constructed from all the passage pairs that have been annotated with the same topic-stance (query). The nodes of the graph represent the individual passages associated with a topic-stance. For a given passage pair (A,B), if passage A is more convincing than passage B (based on the combined assessments from multiple annotators), a directed edge from node A to node B is created. Assuming that every possible passage pair has been annotated, the initial passage graph will be complete.

4 Ranking Model for Convincingness

In this section we describe our base model for predicting argument convincingness, as well as the various approaches for using pairwise-labeled data to train a model for ranking passages. We implement all our models in TensorFlow (Abadi et al., 2016) and tokenize text using NLTK (Bird and Loper, 2004).

4.1 Base Convincingness Model

The base model we use for predicting argument convincingness is an extension of the sum-of-word-embedding approach used by Potash et al. (2017). Their model uses pretrained GloVe word embeddings (Pennington et al., 2014), and, instead of continuing to update the word embedding parameters during training, the model learns a fully-connected layer that projects the embeddings into a new embedding space. By doing so, the original 300-dimensional embeddings are transformed into a 100-dimensional space. The model then sums the projected word embeddings to create a single vector representation of the full passage.[5]

We extend the original model by adding further capacity in the form of a Feed Forward Neural Network (FFNN) after summing the word embeddings. Specifically, we add three additional layers (the original model had a single layer after summing embeddings) of sequentially decreasing size, activated by the ReLU function: these layers have dimensions of 32, 16, 8, and 1. Thus, there is a total of four layers after creating the passage representation, where the last layer produces a single score.

Aside from the strong performance of this model, the fact that it only requires pretrained word embeddings as an external resource makes it appealing, as it increases portability and shortens the preprocessing pipeline. In comparison, the linguistic feature proposed by Habernal and Gurevych (2016) require substantial preprocessing, including part-of-speech tagging, named-entity recognition, and sentiment analysis.

Using the publicly available UKP convincingness dataset from Habernal and Gurevych (2016),

[5]Simple sum-of-word-embeddings has been shown to be a strong (almost unreasonably so) approach for modeling multi-token sequences (Conneau et al., 2017; Joulin et al., 2017).

Model	Pearson's r	Spearman's ρ	Kendall's τ
GPPL (linguistic+word embedding features)	.44	.67	.50
Sum-of-Word-Embeddings+FFNN (our model)	**.48** ($\pm$.013)	**.69** ($\pm$.003)	**.52** ($\pm$.002)

Table 2: Results on the UKP argument convincingness dataset (Habernal and Gurevych, 2016) from our model (Sum-of-Embeddings+FFNN) and Simpson and Gurevych (2018) (GPPL), which had previously been state-of-the-art. Note that our model uses only pretrained word embeddings as features, whereas the GPPL uses pretrained word embeddings plus a linguistic feature space of 32,010. Our numbers are the average across eight identical runs (standard deviation in parentheses).

we test the effectiveness of our base convincingness model against the the current state-of-the-art (Simpson and Gurevych, 2018): Gaussian Process Preference Learning (GPPL) with word embeddings and linguistic features (of dimensionality 32,010) used to represent passages. The evaluation uses a leave-one-*topic*-out paradigm, measures correlation between our model's predictions and the gold standard scores, and averages the correlation scores across topics. Results of our experiments are presented in Table 2 and show that our model achieves a new state-of-the art on the convincingness ranking subtask across all three correlation measures, which were the metrics used by previous researchers on the dataset.

4.2 Methods for Ranking Model

Although Habernal and Gurevych (2016) used PR over directed graphs induced from the pairwise annotations to create unique convincingness scores for single passages within a set, we posit that such a methodology might be sub-optimal for training a ranking model. We address two primary concerns with this approach, and propose solutions, which we detail below.

Train ranking model directly with pairwise data Regressing to any target induced by pairwise-labeled data introduces a system bias based on how the real-valued scores are calculated. It may be better to use the pairwise annotations directly and train with an objective akin to RankNet (Burges et al., 2005). Thus, our base ranking model produces scores independently for each passage in a pair, with the pair of scores then normalized by the softmax function. The softmax outputs become the input probabilities for optimizing a two-class classification function with cross-entropy, where the one-hot target is the argument annotated as more convincing. At test time, our base model then independently produces a global convincingness score for each passage.

Optimize regression based on 'Win-Rate', not PR Assuming we keep the regression objective for training, is there a better way to induce real-valued scores for individual passages? Our training data set, despite its wide topical coverage, only averages four passages per query, with many queries only having two passages. When running PR on a graph with two nodes, directed from one to the other, the node scores become roughly $\frac{2}{3}$ and $\frac{1}{3}$. A simpler, intuitive method for scoring passages would be to assign 1 to the more convincing passage, and 0 to the other. Thus, as an alternative to PR we propose the Win-Rate (WR) of a passage as the regression target. We start with our dataset of passage pairs with a *single* label assigned to the passage that is more convincing (produced by the MACE (Hovy et al., 2013) algorithm taking into account the five raw annotations). We calculate the WR for an individual passage by dividing the number of times a passage is labeled more convincing than another passage by the number of passage pairs it appears in. The scores produced by WR are normalized between 0 and 1 but have a higher variance compared to PR because they do not reflect a probability distribution.

Consequently, we propose to evaluate three different methods of leveraging pairwise-labeled data for training a ranking model: 1) Train directly with pairwise data using a classification objective; 2) Optimize a regression model for WR; 3) Optimize a regression model for PR[6].

5 Experimental Design

In this section we describe the details for evaluating the methods we propose in Section 4.2,

[6]We use the Python package NetworkX (Hagberg et al., 2008) to create graphs and calculate PR scores.

namely the approaches for filtering the fully annotated dataset, as well as creating a properly curated train/test split.

5.1 Creating Train/Test Split

A goal of the convincingness model is to be agnostic to an argument's topic, i.e. the model should perform well on passages even for topics not seen during training. Thus, we create a train/test split not over individual examples, but over topics (where a topic has an associated set of queries, and each query has an associated set of passages). We assign 80% of topics to the training set and the remaining 20% to the test set.

For evaluation, we require gold-standard rankings for passages in a query set. First, we filter the individual examples in the test set by annotator confidence, using a MACE entropy threshold of .95. Next, to ensure no ambiguity in the resulting ranking, we filter all queries that have cycles in their directed passage graphs induced from the pairwise MACE scores[7] (we also remove graphs that have become disconnected due to MACE filtering removing certain edges). To further ensure that the resulting passage rankings are gold-standard, despite not being set-ordered during annotation, we only keep queries whose passage rankings, determined by both WR and longest walk on the passage graphs, are identical. The resulting gold-standard test set contains 659 queries with an average of 2.23 passages per query.

5.2 Filtering/Weighting Training Data

Although the rigorous filtering process for creating the gold-standard test set maintains that the ranks created by sorting on WR generate an unambiguous ordering, doing so reduces the amount of data available. The question then becomes, is it better to keep data with noisy labeling in order to increase the amount of data available for training? In order to evaluate the effect of filtering data in the training set, we experiment with filtering data based on two methods: (1) removing individual annotated passage pairs with MACE entropy score below 0.95[8], and (2) removing query-passage sets if there are cycles present in the passage-graph. Because MACE assigns entropy to each label given to an annotated pair, we also experiment with weighting the training cost of each training example in the pairwise model using its MACE entropy. Specifically, since the passage rated as more convincing has a MACE entropy between 0.5 and 1, we set the training cost weight to $(2{*}entropy){-}1$ producing a weight in the interval (0,1).

6 Results

The results of our experiments are shown in Table 3. For each query-passage set in the test set we predict scores for each passage individually, and evaluate the scores against the gold-standard ranking, as described in Section 5.1. We calculate Kendall's tau and the top 1 accuracy (i.e., the proportion of passage sets where the most convincing passage in the set is ranked first)[9,10]. We average the scores on each query across the test set to produce a single number for each metric. We compare the results of our models with the results of a random baseline and the relevancy score assigned by the search engine to the original *passage*, *query*, *topic* triple (see Section 3).

An initial result of our experiments is that training a pairwise model leads to better ranking performance compared to regressing to a target score for each passage. Furthermore, the use of weighting in training for the pairwise model makes the model more robust with respect to different filtering scenarios of the training data, though we achieve the best correlation with gold standard without using the weighting. Indeed, without weighting during training, the pairwise model only outperforms regression to WR, in terms of correlation to gold standard, in one out of four scenarios of training data filtering. Alternatively, when training models with the complete dataset,

[7] For example, if we have labeled pairs for passages a, b, c, where a is more convincing than b, b is more convincing than c, and c is more convincing than a, then the labeled graph contains a directed cycle.

[8] This process remains the same regardless of whether a model trains on individual passage examples for regression training or passage pairs for pairwise training. However, this procedure affects the amount of data for these types of models differently. For example, given N passages, there are N choose 2 pairs. However, if one pair (edge in the passage graph) is removed due to MACE filtering, there still remains N passages for regression training (assuming the passage graph hasn't become disconnected), but only (N choose 2)-1 passage pairs for pairwise training.

[9] We do not use normalized discounted cumulative gain (nDCG) (Järvelin and Kekäläinen, 2002) because our passage sets are so small. For example, when the set only has two elements, predicting the inverse of the gold-standard still yields an nDCG@2 of 0.63.

[10] Additionally, we do not use Pearson or Spearman correlation, which we used in the UKP experiments, because they are not classical ranking metrics.

Training Objective	Cycles Filtered	MACE Filtered	% Filtered	Kendall's τ	Top1
Pairwise	No	No	0%	.419	.684
Pairwise	No	Yes	43%	.436	.692
Pairwise	Yes	No	61%	**.464***	**.701**
Pairwise	Yes	Yes	48%	.431	.690
Pairwise, Weighted	No	No	0%	.445	.690
Pairwise, Weighted	No	Yes	43%	.451	.701
Pairwise, Weighted	Yes	No	61%	**.458**	**.704***
Pairwise, Weighted	Yes	Yes	48%	.455	.700
Regression to PR	No	No	0%	.408	**.677**
Regression to PR	No	Yes	13%	**.411**	.676
Regression to PR	Yes	No	40%	.392	.657
Regression to PR	Yes	Yes	18%	.399	.669
Regression to WR	No	No	0%	.442	.688
Regression to WR	No	Yes	13%	.445	.692
Regression to WR	Yes	No	40%	**.456**	**.695**
Regression to WR	Yes	Yes	18%	.431	.684
Random	-	-	-	.000	.447
Relevancy Ranking	-	-	-	.204	.585

Table 3: Results of ranking experiments on our newly-annotated dataset. Bold indicates the best performance for a given model on a given evaluation metric, and * indicates the best result across all models.

i.e. not using any filtering, regressing to WR is better than pairwise training *without* weighting.

In terms of regression targets, WR is shown to be a superior objective compared to PR. Furthermore, this holds across all variations of filtering the training data. In fact, PR exhibits its worst performance under the filtering constraints where WR performs the best. These results show that even if one has decided on a regression objective, the way in which one calculates the scores to which the model fits is important.

When examining the effects of data filtering, combining strategies is not always better. Our results show that it is better to filter out whole passage sets that have cycles, as opposed to filtering out individual examples based on MACE score. However, if MACE filtering has already been done, it is generally better to leave cycle-inducing passage sets in the training data. These results indicate that there may be a fine line between removing noise and removing useful information. There is also an interesting relationship between MACE filtering and cycle filtering. We observe that filtering for cycles after initially filtering by MACE results in *more* data being left, when compared with solely filtering by cycles. This implies that MACE entropy scores are able to predict which labels may lead to cycles in a passage graph.

6.1 Convincingness versus Relevancy

Regarding the actual utility of ranking passages by argument convincingness, as opposed to just using topical relevancy, our results show that in fact convincingness and relevancy are separate attributes when it comes to grading a passage. Although the use of relevancy ranking scores results in more convincing passages being selected than random guessing, the relevancy model does not predict argument convincingness as effectively as a model trained specifically to do so. In other words, when constructing a search engine for arguments, the most topically relevant passage may not be the most convincing with regard to its stance on an issue. Future work can evaluate the best practice for combining these different attributes for the best user experience.

7 Conclusion

Our work provides a practical case study in the use of pairwise-annotated data to train a model for ranking passages with respect to their argumentative convincingness. We describe an annotation process that takes the raw output of a search engine and transforms the data into pairs of pas-

sages with the same stance toward an issue, annotated for which passage is more convincing. We then construct a base model for predicting argument convincingness that posts state-of-the-art on a publicly available dataset. We conclude with a comprehension evaluation of different ranking models using our newly-annotated dataset. Our results show that a pairwise model trained with cross-entropy objective provides the best performance, though regressing to a simple Win-Rate target can also perform competitively.

Acknowledgments

We would like to thank Frank Guo and Xuan Li for providing the initial seed data for our annotation, Bhaskar Mitra for discussing approaches for ranking methods, and Tong Wang for his help reviewing the manuscript.

References

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. 2016. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283.

Héctor Martínez Alonso, Barbara Plank, Arne Skjærholt, and Anders Søgaard. 2015. Learning to parse with iaa-weighted loss. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1357–1361.

Pablo Barberá, John T Jost, Jonathan Nagler, Joshua A Tucker, and Richard Bonneau. 2015. Tweeting from left to right: Is online political communication more than an echo chamber? *Psychological science*, 26(10):1531–1542.

Eyal Beigman and Beata Beigman Klebanov. 2009. Learning with annotation noise. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 280–287. Association for Computational Linguistics.

Steven Bird and Edward Loper. 2004. Nltk: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*, page 31. Association for Computational Linguistics.

Ralph Allan Bradley and Milton E Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345.

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. 2005. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96. ACM.

Lisa Andreevna Chalaguine and Claudia Schulz. 2017. Assessing convincingness of arguments in online debates with limited number of features. In *Proceedings of the Student Research Workshop at the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 75–83.

Xi Chen, Paul N Bennett, Kevyn Collins-Thompson, and Eric Horvitz. 2013. Pairwise ranking aggregation in a crowdsourced setting. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 193–202. ACM.

Yuxin Chen and Changho Suh. 2015. Spectral mle: Top-k rank aggregation from pairwise comparisons. In *International Conference on Machine Learning*, pages 371–380.

Wei Chu and Zoubin Ghahramani. 2005. Preference learning with gaussian processes. In *Proceedings of the 22nd international conference on Machine learning*, pages 137–144. ACM.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680.

Esin Durmus and Claire Cardie. 2018. Exploring the role of prior beliefs for argument persuasion. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1035–1045.

Yoav Freund and Robert E Schapire. 1999. Large margin classification using the perceptron algorithm. *Machine learning*, 37(3):277–296.

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 (Volume 1: Long Papers)*, volume 1, pages 1589–1599.

Aric Hagberg, Pieter Swart, and Daniel S Chult. 2008. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Sandra Heldsinger and Stephen Humphry. 2010. Using the method of pairwise comparison to obtain reliable teacher assessments. *The Australian Educational Researcher*, 37(2):1–19.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with mace. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130.

Kathleen Hall Jamieson and Joseph N Cappella. 2008. *Echo chamber: Rush Limbaugh and the conservative media establishment*. Oxford University Press.

Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, volume 2, pages 427–431.

Peter John Loewen, Daniel Rubenson, and Arthur Spirling. 2012. Testing the power of arguments in referendums: A bradley–terry approach. *Electoral Studies*, 31(1):212–221.

Sahand Negahban, Sewoong Oh, and Devavrat Shah. 2016. Rank centrality: Ranking from pairwise comparisons. *Operations Research*, 65(1):266–287.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 742–751.

Peter Potash, Robin Bhattacharya, and Anna Rumshisky. 2017. Length, interchangeability, and external knowledge: Observations from predicting argument convincingness. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 342–351.

Nihar B Shah, Sivaraman Balakrishnan, Joseph Bradley, Abhay Parekh, Kannan Ramchandran, and Martin Wainwright. 2014. When is it better to compare than to score? *arXiv preprint arXiv:1406.6618*.

Edwin D Simpson and Iryna Gurevych. 2018. Finding convincing arguments using scalable bayesian preference learning. *Transactions of the Association for Computational Linguistics*, 6:357–371.

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: Demonstrations*, pages 21–25.

Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. In *Proceedings of the 25th international conference on world wide web*, pages 613–624. International World Wide Web Conferences Steering Committee.

Louis L Thurstone. 1927. A law of comparative judgment. *Psychological review*, 34(4):273.

Cristian Vaccari. 2013. From echo chamber to persuasive device? rethinking the role of the internet in campaigns. *New Media & Society*, 15(1):109–127.

Kevin Wallsten. 2005. Political blogs and the bloggers who blog them: Is the political blogosphere and echo chamber. In *American Political Science Associations Annual Meeting. Washington, DC September*, pages 1–4.

Elad Yom-Tov, Susan Dumais, and Qi Guo. 2014. Promoting civil discourse through search engine diversity. *Social Science Computer Review*, 32(2):145–154.