Automated Controversy Detection on the Web

Shiri Dori-Hacohen and James Allan

Center for Intelligent Information Retrieval
School of Computer Science
University of Massachusetts, Amherst, Amherst MA 01002, USA,
{shiri,allan}@cs.umass.edu

Abstract. Alerting users about controversial search results can encourage critical literacy, promote healthy civic discourse and counteract the "filter bubble" effect, and therefore would be a useful feature in a search engine or browser extension. In order to implement such a feature, however, the binary classification task of determining which topics or webpages are controversial must be solved. Earlier work described a proof of concept using a supervised nearest neighbor classifier with access to an oracle of manually annotated Wikipedia articles. This paper generalizes and extends that concept by taking the human out of the loop, leveraging the rich metadata available in Wikipedia articles in a weakly-supervised classification approach. The new technique we present allows the nearest neighbor approach to be extended on a much larger scale and to other datasets. The results improve substantially over naive baselines and are nearly identical to the oracle-reliant approach by standard measures of F_1, $F_{0.5}$, and accuracy. Finally, we discuss implications of solving this problem as part of a broader subject of interest to the IR community, and suggest several avenues for further exploration in this exciting new space.

1 Introduction

On the web today, alternative medicine sites appear alongside pediatrician advice websites, the phrase "global warming is a hoax" is in wide circulation, and political debates rage in many nations over economic issues, same-sex marriage and healthcare. Access does not translate into trustworthy information: e.g., parents seeking information about vaccines will find plenty of "proof" that they cause autism, and may not even realize the depth of the controversy involved [1]; ads for helplines displayed to users searching for "abortion" are discreetly funded by pro-life (anti-abortion) religious groups [10]. The underlying thread connecting all these examples is that users searching for these topics may not even be aware that a controversy exists; indeed, without the aid of a search engine feature or browser extension to warn them, they may never find out. We believe that informing users about controversial topics would be a valuable addition to the end-user experience; this requires detecting such topics as a prerequisite.

In prior work, we analyzed whether the structural properties of the problem allow for a solution by proxy via Wikipedia, and demonstrated that there is a correlation between controversiality of Wikipedia pages and that of the

webpages related to them [7]. We performed a proof-of-concept upper-bound analysis, using human-in-the-system judgments as an oracle for the controversy level of related Wikipedia articles. This naturally raises the question of whether an actual controversy detection system for the web can be constructed, making use of these properties.

In this work, we are putting these insights to use by introducing a novel, fully-automated system for predicting that arbitrary webpages discuss controversial topics. Our contribution is a weakly-supervised approach to detect controversial topics on arbitrary web pages. We consider our system as distantly-supervised [16] since we use heuristic labels for neighboring Wikipedia articles, which act as a bridge between the rich metadata available in Wikipedia and the sparse data on the web. One might hypothesize that using an automated system to scoring Wikipedia articles (instead of an oracle of human annotations) would degrade the results. In fact, however, our approach achieves comparable results to the prior art, which represented an upper-bound on this approach [7], while at the same time making it applicable to any large-scale web dataset.

2 Related Work

Several strands of related work inform our work: controversy detection in Wikipedia, controversy on the web and in search, fact disputes and trustworthiness, as well as sentiment analysis. We describe each area in turn.

Controversy detection in Wikipedia. Several papers focused on detecting controversy in Wikipedia [12, 17, 21], largely using metadata features such as length of the talk page, proportion of anonymous editors, and certain types of edits such as reverts. We describe a few of these in more detail in Section 3.2. Wikipedia is a valuable resource, but often "hides" the existence of debate by presenting even controversial topics in deliberately neutral tones [20], which may be misleading to people unfamiliar with the debate.

While detecting controversy in Wikipedia automatically can be seen as an end in itself, these detection methods have wider reach and can be used as a step for solving other problems. Recently, Das et al. used controversy detection as a step to study manipulation by Wikipedia administrators [6]. Additionally, Wikipedia has been used in the past as a valuable resource assisting in controversy detection elsewhere, whether as a lexicon or as a hierarchy for controversial words and topics [3, 15]. Likewise, we use a few of the Wikipedia-specific controversy measures described above as a step in our approach (see Section 3.2).

As described above, prior work showed an upper-bound analysis demonstration using related Wikipedia articles as a proxy for controversy on the web, by using human annotations as an oracle rating the controversy of the articles [7]. In contrast, we use automatically-generated values for the Wikipedia articles.

Controversy on the web and in search. Outside of Wikipedia, other targeted domains such as news [3, 5] and Twitter [15] have been mined for controversial topics, mostly focusing on politics and politicians. Some work relies on domain-specified sources such as Debatepedia[1] [3, 11] that are likewise politics-

[1] http://dbp.idebate.org/

heavy. We consider controversy to be wider in scope; medical and religious controversies are equally interesting. A query completion approach might be useful in detecting controversial queries [9]; assuming one knows that a query is controversial, diversifying search results based on opinions is a useful feature [11].

Fact disputes and trustworthiness are often related to controversial topics [8, 19]. Similar to our goal, the Dispute Finder tool focused on finding and exposing disputed claims on the web to users as they browse [8]. However, Dispute Finder was focused on manually added or bootstrapped fact disputes, whereas we are interested in scalably detecting controversies that may stem from fact disputes, but also from disagreement on values or from moral debates.

Sentiment analysis can naturally be seen as a useful tool as a step towards detecting varying opinions, and potentially controversy [5, 15, 18]. However, as mentioned elsewhere [3, 7], sentiment alone may not suffice for detecting controversy, though it may be useful as a feature.

3 Nearest Neighbor approach

Our approach to detecting controversy on the web is a nearest neighbor classifier that maps webpages to the Wikipedia articles related to them. We start from a webpage and find Wikipedia articles that discuss the same topic; if the Wikipedia articles are controversial, it is reasonable to assume the webpage is controversial as well. Prior work demonstrated that this approach worked using human judgment [7], leaving open the question of whether a fully-automated approach can succeed.

The choice to map specifically to Wikipedia rather than to any webpages was driven by the availability of the rich metadata and edit history on Wikipedia [12, 17, 21]. We consider our approach as a distantly-supervised classifier in the relaxed sense (c.f. [16]), since we are using automatically-generated labels, rather than truth labels, for an external dataset (Wikipedia) rather than the one we are training on (web). While some of these labels were learned using a supervised classifier on Wikipedia, none of them were trained for the task at hand, namely classifying webpages' controversy.

To implement our nearest neighbor classifier, we use several modules: matching via query generation, scoring the Wikipedia articles, aggregation, thresholding and voting. We describe each in turn.

3.1 Matching via Query Generation

We use a query generation approach to map from webpages to the related Wikipedia articles. The top ten most frequent terms on the webpage, excluding stop words, are extracted from the webpage, and then used as a keyword query restricted to the Wikipedia domain and run on a commercial search engine. We use one of two different stop sets, a 418 word set (which we refer to as "Full" Stopping [4]) or a 35 word set ("Light" Stopping [13]). Wikipedia redirects were followed wherever applicable in order to ensure we reached the full Wikipedia article with its associated metadata; any talk or user pages were ignored.

We considered the articles returned from the query as the webpage's "neighbors", which will be evaluated for their controversy level. Based on the assumption that higher ranked articles might be more relevant, but provide less coverage, we varied the number of neighbors in our experiments from 1 to 20, or used all articles containing all ten terms. A brief evaluation of the query generation approach is presented in Section 5.1.

3.2 Automatically-generated Wikipedia labels

The Wikipedia articles, found as neighbors to webpages, were labeled with several scores measuring their controversy level. We use three different types of automated scores for controversy in Wikipedia, which we refer to as **D**, **C**, and **M** scores. All three scores are automatically generated based on information available in the Wikipedia page and its associated metadata, talk page and revision history. While we use a supervised threshold on the scores, the resulting score and prediction can be generated with no human involvement.

The D score tests for the presence of **D**ispute tags that are added to the talk pages of Wikipedia articles by its contributors [12, 17]. These tags are sparse and therefore difficult to rely on [17], though potentially valuable when they are present. We test for the presence of such tags, and use the results as a binary score (1 if the tag exists or -1 if it doesn't). Unfortunately, the number of dispute tags available is very low: in a recent Wikipedia dump, only 0.03% of the articles had a dispute tag on their talk page. This is an even smaller dataset than the human annotations provided in prior work [7]; the overlap between these articles and the 8,755 articles in the dataset is a mere 165 articles.

The C score is a metadata-based regression that predicts the controversy level of the Wikipedia article using a variety of metadata features (e.g. length of the page and its associated talk page, number of editors and of anonymous editors). This regression is based on the approach first described by Kittur et al. [12]. We use the version of this regression as implemented and trained recently by Das et al. [6], generating a floating point score in the range (0,1).

The M score, as defined by Yasseri et al., is a different way of estimating the controversy level of a Wikipedia article, based on the concept of mutual reverts and edit wars in Wikipedia [21]. Their approach is based on the number and reputation of the users involved in reverting each others' edits, and assumes that "the larger the armies, the larger the war" [21]. The score is a positive real number, theoretically unbounded (in practice it ranges from 0 to several billion).

3.3 Aggregation and Thresholding

The score for a webpage is computed by taking either the maximum or the average of all its Wikipedia neighbors' scores, a parameter we vary in our experiments. After aggregation, each webpage has 3 "controversy" scores from the three scoring methods (**D**, **C** and **M**). We trained various thresholds for both **C** and **M** (see Section 4.1), depending on target measures.

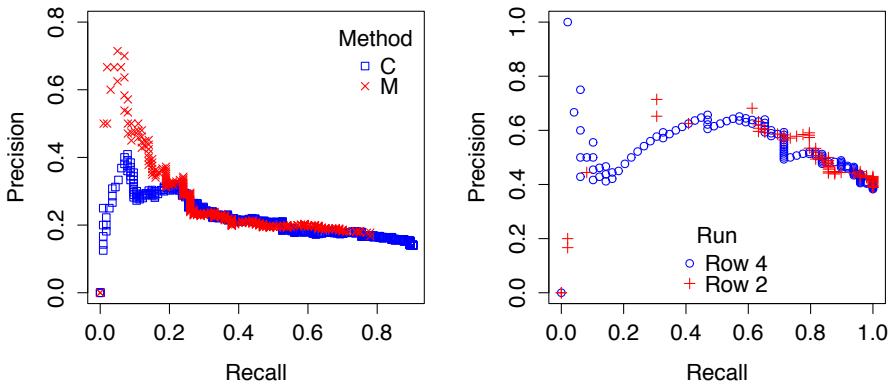

Fig. 1: Precision-Recall curves (uninterpolated). Left: PR curve for C and M thresholds on the Wikipedia NNT set. Right: PR curve for select runs on the Test set. Row numbers refer to Table 2.

3.4 Voting

In addition to using each of the three labels in isolation, we can also combine them by voting. We apply one of several voting schemes to the binary classification labels, after the thresholds have been applied. The schemes we use are:

- Majority vote: consider the webpage controversial if at least two out of the three labels are "controversial".
- Logical *Or*: consider the webpage controversial if any of the three labels is "controversial".
- Logical *And*: consider the webpage controversial only if all the three labels are "controversial".
- Other logical combinations: we consider results for the combination $(Dispute \vee (C \wedge M))$, based on the premise that if the dispute tag happens to be present, it would be valuable[2].

4 Experimental Setup and Data Set

To compare to prior work, we use the dataset used in previous experiments [7], consisting of webpages and Wikipedia articles annotated as controversial or non-controversial. This publicly-released dataset includes 377 webpages, and 8,755 Wikipedia articles. Of the Wikipedia articles annotated in the set, 4,060 were the Nearest Neighbors associated with the Training set ("NNT" in Table 1), which we use later (see Section 4.1). For evaluation, we use Precision, Recall, Accuracy,

[2] D's coverage was so low that other voting combinations were essentially identical to the majority voting; we therefore omit them.

Webpages		
Set	Pages	Controversial
All	377	123 (32.6%)
Training	248	74 (29.8%)
Testing	129	49 (38.0%)

Wikipedia articles			
Set	Articles	Annotated	Controversial
All	8,755	1,761	282 (16.0%)
NNT	4,060	853	115 (13.5%)

Table 1: Data set size and annotations. "NNT" denotes the subset of Wikipedia articles that are Nearest Neighbors of the webpages Training set.

F_1 and $F_{0.5}$ using the classic IR sense of these metrics, with "controversial" and "non-controversial" standing in for "relevant" and "non relevant", respectively.

4.1 Threshold training

C and **M** are both real-valued numbers; in order to generate a binary classification, we must select a threshold above which the page will be considered controversial. (**D** score is already binary.) Since the public corpus has annotations on some of the Wikipedia articles [7], we trained the thresholds for **C** and **M** for the subset of articles associated with the training set (labeled "NNT" in Table 1). The Precision-Recall curve for both scores is displayed in Figure 1. We select five thresholds for the two scoring methods, based on the best results achieved on this subset for our measures.

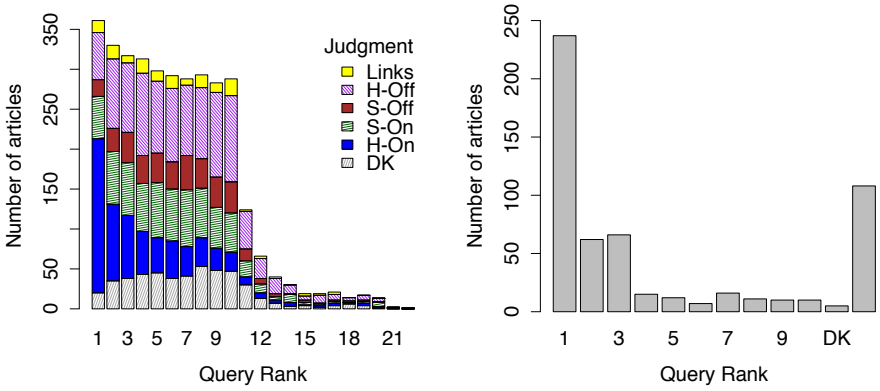

Fig. 2: Evaluation of Matching scheme. Left: Judgments on Wikipedia articles returned by the automatically-generated queries, by rank. Annotators could choose one of the following options: H-On="Highly on [webpage's] topic", S-On="Slightly on topic", S-Off="Slightly off topic", H-Off="Highly off topic", Links="Links to this topic, but doesn't discuss it directly", DK="Don't Know". Right: Frequency of page selected as best, by rank. DK="Don't Know", N="None of the above".

For comparison, we also present single-class acceptor baselines on this task of labeling the Wikipedia articles, one which labels all pages as non-controversial and one which labels all pages as controversial. Finally, two random baselines which label every article as either controversial or non-controversial based on a coin flip, are presented for comparison (average of three random runs). One of these baselines flips a coin with 50% probability, and the other flips it with 29.8% probability (the incidence of controversy in the training set).

5 Evaluation

We treat the controversy detection problem as a binary classification problem of assigning labels of "controversial" and "non-controversial" to webpages. We present a brief evaluation for the query generation approach before turning to describe our results for the controversy detection problem.

5.1 Judgments from Matching

A key step in our approach is selecting which Wikipedia articles to use as nearest neighbors. In order to evaluate how well our query generation approach is mapping webpages to Wikipedia articles, we evaluated the automated queries and the relevance of their results to the original webpage. This allows an intrinsic measure of the effectiveness of this step - independent of its effect on the extrinsic task, which is evaluated using the existing dataset's judgments on the webpages' controversy level[3]. We annotated 3,430 of the query-article combinations (out of 7,630 combinations total) that were returned from the search engine; the combinations represented 2,454 unique Wikipedia articles. Our annotators were presented with the webpage and the titles of up to 10 Wikipedia articles in alphabetical order (not ranked); they were not shown the automatically-generated query. The annotators were asked to name the single article that best matched the webpage, and were also asked to judge, for each article, whether it was relevant to the original page. Figure 2 shows how the ranked list of Wikipedia articles were judged. In the figure, it is clear that the top-ranking article was viewed as highly on topic but then the quality dropped rapidly. However, if both "on-topic" judgments are combined, a large number of highly or slightly relevant articles are being selected. Considering the rank of the best article as the single relevant result, the Mean Reciprocal Rank for the dataset was 0.54 (if the best article was "don't know" or "none of the above", its score was zero).

5.2 Our results compared to baseline runs

We compare our approach to several baselines, a sentiment analysis approach based on a logistic regression classifier [2] trained to detect presence of sentiment on the webpage, whether positive or negative; sentiment is used as a proxy for controversy. We add single-class and random baselines (average of three runs).

[3] Both sets are publicly released - see http://ciir.cs.umass.edu/downloads

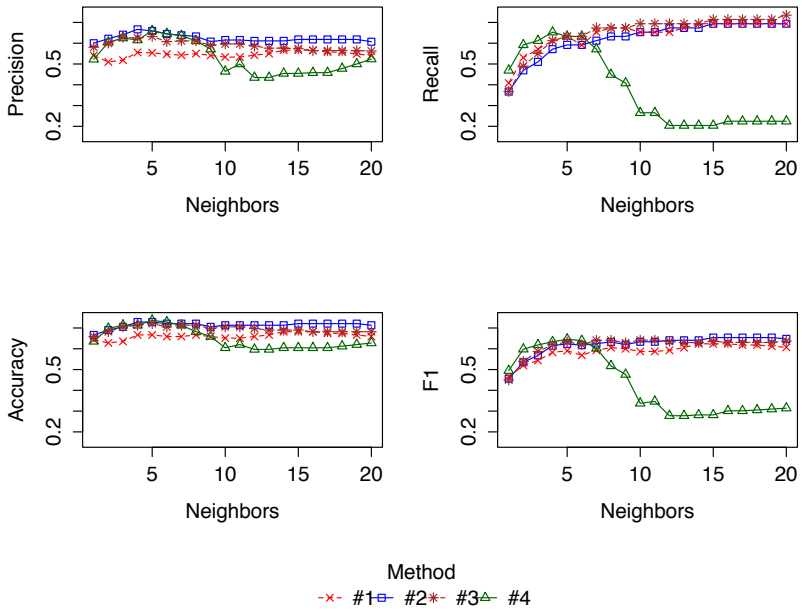

Fig. 3: Evaluation metrics vary with number of neighbors (k). Refer to rows 1-4 in Table 2: rows 1 and 4 use avg aggregator with low thresholds, while 2 and 3 use max with high thresholds.

Finally, the best results from our prior work [7] are reported. As described in Section 3, we varied several parameters in our nearest neighbor approach:

1. **Stopping set** (Light or Full)
2. **Number of neighbors** (k=1..20, or no limit)
3. **Aggregation method** (average or max)
4. **Scoring or voting method** (C, M, D; Majority, Or, And, $D \vee (C \wedge M)$)
5. **Thresholds for C and M** (one of five values, as described in Section4.1).

These parameters were evaluated on the training set and the best runs were selected, optimizing for F_1, $F_{0.5}$ and Accuracy. The parameters that performed best, for each of the scoring/voting methods, were then run on the test set.

The results of our approach on the test set are displayed in Table 2. For ease of discussion, we will refer to row numbers in the table. For space considerations, highly similar runs are omitted.

6 Discussion

As the results in Table 2 show, our fully-automated approach (rows 1-11) achieves results higher than all baselines (rows 15-19), in all metrics except recall (which is trivially 100% in row 19). The method that optimized for $F_{0.5}$ on the training

Table 2: Results on Testing Set. Results are displayed for the best parameters on the training set, using each scoring method, optimized for F_1, Accuracy and $F_{0.5}$. The overall best results of our runs, in each metric, are displayed in bold; the best prior results (rows 12-14 [7]) and baseline results (rows 15-19) are also displayed in bold. See text for discussion.

	Parameters							Test Metric				
#	Stop	Score	k	agg	Thres C	Thres M	Target	P	R	F_1	Acc	$F_{0.5}$
1	Full	M	8	avg	–	*84930*	F_1, Acc	0.55	0.67	0.61	0.67	0.57
2	Light	M	8	max	–	*2.85*$\times 10^6$	$F_{0.5}$	0.63	0.63	0.63	**0.72**	**0.63**
3	Light	C	15	max	*0.17*	–	F_1	0.57	0.71	0.64	0.69	0.60
4	Light	C	7	avg	*4.18*$\times 10^{-2}$	–	Acc, $F_{0.5}$	**0.64**	0.57	0.60	0.71	0.62
5	Light	D	19	max	–	–	F_1	0.43	0.57	0.49	0.55	0.45
6	Full	D	5	max	–	–	Acc	0.53	0.37	0.43	0.64	0.49
7	Light	D	6	max	–	–	Acc, $F_{0.5}$	0.44	0.35	0.39	0.58	0.41
8	Light	Maj.	15	max	*0.17*	*2.85*$\times 10^6$	F_1	0.59	0.73	**0.65**	0.70	0.61
9	Full	Maj.	5	max	*4.18*$\times 10^{-2}$	*2.85*$\times 10^6$	Acc, $F_{0.5}$	0.59	0.61	0.60	0.69	0.59
10	Light	And	no	max	*0.17*	*84930*	F_1, Acc, $F_{0.5}$	0.52	0.51	0.51	0.64	0.52
11	Light	D|CM	7	avg	*4.18*$\times 10^{-2}$	*84930*	Acc, $F_{0.5}$	0.63	0.55	0.59	0.70	0.61
12	Oracle-based [7], best run for P, Acc and $F_{0.5}$							**0.69**	0.51	0.59	**0.73**	**0.65**
13	Oracle-based [7], best run for R							0.51	**0.84**	0.64	0.64	0.56
14	Oracle-based [7], best run for F_1							0.60	0.69	**0.64**	0.70	0.61
15	Sentiment [7]							0.38	0.90	0.53	0.40	0.43
16	Random_{50}							**0.42**	0.53	0.47	0.54	**0.44**
17	$\text{Random}_{29.8}$							0.23	0.19	0.21	0.61	0.22
18	All non-controversial							0	0	0	**0.62**	0
19	All Controversial							0.38	**1.00**	**0.55**	0.38	0.43

set among all the single score approaches was the run using Light Stopping, M with a rather high (discriminative) threshold, and aggregating over the maximal value of all the result neighbors (row 2 in Table 2). Using k values of 8 through 12 achieved identical results on the training set. These runs ended up achieving some of the best results on the test set; with value k=8 the results were the best for $F_{0.5}$ as well as Accuracy (row 2), with 10.1% absolute gain in accuracy (16.3% relative gain) over the non-controversial class baseline, which had the best accuracy score among the baselines. For $F_{0.5}$ this run showed 19.5% absolute gain (44.5% relative gain) over the best $F_{0.5}$ score, which was achieved by the Random_{50} baseline. Even though none of the results displayed in the table were optimized for precision, they still had higher precision than the baselines across the board (compare rows 1-11 to rows 15-19). Among the voting methods, the method that optimized for F_1 on the training set was the Majority voting, using Light Stopping, aggregating over the maximal value of 15 neighbors, with discriminative thresholds for both M and C (row 12). This run showed a 10.4% (18.9% relative gain) absolute gain on the test set over the best baseline for F_1.

The results of the sentiment baseline (row 15) were surprisingly similar to a trivial acceptor of "all controversial" baseline (row 19); at closer look, the sentiment classifier only returns about 10% of the webpages as lacking sentiment, and thus its results are close to the baseline. We tried applying higher confidence thresholds to the sentiment classifier, but this resulted in lower recall without improvement in precision. We note that the sentiment classifier was not trained to detect controversy; it's clear from these results, as others have noted, that sentiment alone is too simplistic to predict controversy [3, 7].

When comparing our results (rows 1-11) to the best oracle-reliant runs from prior work (rows 12-14, see [7]), the results are quite comparable. Recall that this prior work represents a proof-of-concept upper-bound analysis, with a human-in-the-loop providing judgments for the relevant Wikipedia pages, rather than an automatic system that can be applied to arbitrary pages[4]. When comparing the best prior work result (row 12) to our best run (row 2) using a zero-one loss function, the results were not statistically different. This demonstrates that our novel, fully-automated system for detecting controversy on the web is as effective as upper-bound, human-mediated predictions [7].

We observe that when using a max aggregator, results were generally better with more discriminative thresholds and a large number of neighbors (k); when average was used, a lower threshold with smaller k was more effective. To understanding this phenomenon, we fixed all the parameters from rows 1-4 above except for k, and plotted system results as a function of k (see Figure 3). Consider that the max function is more sensitive to noise than the average function - a higher threshold can reduce the sensitivity to such noise while extending coverage by considering more neighbors. In most runs depicted, precision drops a little but remains fairly consistent with k, while recall increases steadily. However, in the parameters from row 4, there is a penalty to both precision and recall as k increases, demonstrating the noise sensitivity of the max function.

7 Conclusions and Future Work

We presented the first fully automated approach to solving the recently proposed binary classification task of web controversy detection [7]. We showed that such detection can be performed by automatic labeling of exemplars in a nearest neighbor classifier. Our approach improves upon previous work by creating a scalable distantly-supervised classification system, that leverages the rich metadata available in Wikipedia, using it to classify webpages for which such information is not available. We reported results that represent 20% absolute gains in F measures and 10% absolute gains in accuracy over several baselines, and are comparable to prior work that used human annotations as an oracle [7].

[4] Note that this is not a strict upper-bound limit in the theoretical sense, but in principle it's reasonable to assume that a human annotator would perform as well as an automated system. In fact, in a few cases the automated system performed better than the oracle-reliant approach, see e.g. F1 on row 8 vs. row 14.

Our approach is modular and therefore agnostic to the method chosen to score Wikipedia articles; like Das et al. [6], we can leverage future improvements in this domain. For example, scores based on a network collaboration approach [17] could be substituted in place of the **M** and **C** values, or added to them as another feature. The nearest neighbor method we described is also agnostic to the choice of target collection we query; other rich web collections which afford controversy inference, such as Debate.org, Debatabase or procon.org, could also be used to improve precision.

Future work could improve on our method: better query generation methods could be employed to match neighbors, using entity linking for Wikification could create the links directly, or else language models could compare candidate neighbors directly. Standard machine learning approaches can be used to combine our method with other features such as sentiment analysis.

The nearest neighbor approach we presented is limited in nature by the collection it targets; it will not detect controversial topics that are not covered by Wikipedia. Entirely different approaches would need to be employed to detect such smaller controversies. Nonetheless, it's possible that some metric of sentiment variance across multiple websites could provide useful clues. Another approach could use language models or topic models to automatically detect the fact that strongly opposing, biased points of view exist on a topic, and thus it is controversial. This would flip the directionality of some recent work that presupposes subjectivity and bias to detect points of view [6, 22].

We see the controversy detection problem as a prerequisite to several other interesting applications and larger problems such as: user studies on the effects of informing users when the webpage they are looking at is controversial; the evolution and incidence of controversial topics over time; and diversifying controversial search results according to the stances on them, are a few such problems.

With the growing trend towards personalization in search comes a risk of fragmenting the web into separate worlds, with search engines creating a self-fulfilling prophecy of users' bias confirmation. Informing users about fact disputes and controversies in their queries can improve trustworthiness in search; explicitly exposing bias and polarization may partially counteract the "filter bubble" or "echo chamber" effects, wherein click feedback further reenforce users' predispositions. Further development and refinement of controversy detection techniques can foster healthy debates on the web, encourage civic discourse, and promote critical literacy for end-users of search.

Acknowledgments. Our thanks go to Allen Lavoie, Hoda Sepehri Rad, Taha Yasseri and Elad Yom-Tov for valuable resources and fruitful discussions. Thanks to Sandeep Kalra, Nada Naji, Ravali Pochampally, Emma Tosch, David Wemhoener, Celal Ziftci, and anonymous reviewers for comments on various drafts. Special thanks go to Gonen Dori-Hacohen, without whose valuable and timely assistance, this paper would not have been possible. This work was supported in part by the Center for Intelligent Information Retrieval and in part by NSF grant #IIS-1217281. Any opinions, findings and conclusions or recommendations expressed in this material are the authors, and do not necessarily reflect those of the sponsor.

References

1. Activist Post. 22 Medical Studies That Show Vaccines Can Cause Autism. Accessed from http://www.activistpost.com/2013/09/22-medical-studies-that-show-vaccines.html on Sept. 24, 2014.
2. E. Aktolga and J. Allan. Sentiment Diversification With Different Biases. In *Proc. of SIGIR'13*, pages 593–602, 2013.
3. R. Awadallah, M. Ramanath, and G. Weikum. Harmony and Dissonance: Organizing the People's Voices on Political Controversies. In *Proc. of WSDM '12*, pages 523–532, Feb. 2012.
4. J. P. Callan, W. B. Croft, and S. M. Harding. The INQUERY retrieval system. In *Database and Expert Systems Applications*, pages 78–83. Springer Vienna, 1992.
5. Y. Choi, Y. Jung, and S.-H. Myaeng. Identifying Controversial Issues and Their Sub-topics in News Articles. *Intelligence and Security Informatics*, 6122:140–153, 2010.
6. S. Das, A. Lavoie, and M. Magdon-Ismail. Manipulation Among the Arbiters of Collective Intelligence: How Wikipedia Administrators Mold Public Opinion. In *Proc. of CIKM '13*, pp. 1097–1106, 2013.
7. S. Dori-Hacohen and J. Allan. Detecting controversy on the web. In *Proc. of CIKM '13*, pp. 1845–1848, 2013.
8. R. Ennals, B. Trushkowsky, and J. M. Agosta. Highlighting disputed claims on the web. In *Proc. of WWW '10*, p. 341, 2010.
9. K. Gyllstrom and M.-F. M. Moens. Clash of the typings: finding controversies and children's topics within queries. In *Proc. of ECIR'11*, pp. 80–91, 2011.
10. Heroic Media. Free Abortion Help website, Jan. 2014. Accessed from http://freeabortionhelp.com/us/ on Sept. 24, 2014.
11. M. Kacimi and J. Gamper. MOUNA: Mining Opinions to Unveil Neglected Arguments. In *Proc. of CIKM '12*, p. 2722.
12. A. Kittur, B. Suh, B. A. Pendleton, E. H. Chi, L. Angeles, and P. Alto. He Says, She Says: Conflict and Coordination in Wikipedia. In *Proc. of CHI '07*, pp. 453–462, 2007.
13. C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge, 2008.
14. E. Pariser. *The Filter Bubble: What the Internet is hiding from you*. Penguin Press HC, 2011.
15. A. A.-M. Popescu and M. Pennacchiotti. Detecting controversial events from twitter. In *Proc. CIKM '10*, pp. 1873–1876, 2010.
16. S. Riedel, L. Yao, and A. McCallum. Modeling Relations and Their Mentions Without Labeled Text. In *Proc. ECML PKDD'10*, pp. 148–163.
17. H. Sepehri Rad and D. Barbosa. Identifying controversial articles in Wikipedia: A comparative study. In Proc. WikiSym, 2012.
18. M. Tsytsarau, T. Palpanas, and K. Denecke. Scalable detection of sentiment-based contradictions. *DiversiWeb 2011*, 2011.
19. V. G. V. Vydiswaran, C. Zhai, D. Roth, and P. Pirolli. BiasTrust: Teaching Biased Users About Controversial Topics. In Proc. CIKM '12, pp. 1905–1909, 2012.
20. Wikipedia. Wikipedia: Neutral Point of View Policy, Jan. 2014.
21. T. Yasseri, R. Sumi, A. Rung, A. Kornai, and J. Kertész. Dynamics of conflicts in Wikipedia. *PloS one*, 7(6):e38869, Jan. 2012.
22. E. Yom-Tov, S. T. Dumais, and Q. Guo. Promoting civil discourse through search engine diversity. *Social Science Computer Review*, 2013.