Probabilistic Approaches to Controversy Detection

Myungha Jang, John Foley, Shiri Dori-Hacohen, and James Allan
Center for Intelligent Information Retrieval
College of Information and Computer Sciences
University of Massachusetts
{mhjang, jfoley, shiri, allan}@cs.umass.edu

ABSTRACT

Recently, the problem of automated controversy detection has attracted a lot of interest in the information retrieval community. Existing approaches to this problem have set forth a number of detection algorithms, but there has been little effort to model controversy directly.

In this paper, we propose a probabilistic framework to detect controversy on the web, and investigate two models. We first recast a state-of-the-art controversy detection algorithm into a model in our framework. Based on insights from social science research, we also introduce a language modeling approach to this problem. We extensively evaluate different methods of creating *controversy language models* based on a diverse set of public datasets including Wikipedia, Web and News corpora.

Our automatically derived language models show a significant relative improvement of 18% in AUC over prior work, and 23% over two manually curated lexicons.

1. INTRODUCTION

The power of search is inherently coupled to the trustworthiness and reliability of the dataset the user is searching. Web search can be both be helpful and troubling in this regard. In medicine, one may find misinformation that professionals consider incorrect, whether it is fraudulent treatments, or side-effects that simply do not exist, such as the link between vaccines and autism. In the political domain, increasing polarization of sources and viewpoints leads to "Filter Bubbles" [14], where users are only able to find results that agree with them or the way they formulated their query. In these domains and more, the ability to reliably detect the presence of controversy in search results could aid critical literacy and the ability to make real-life decisions. Our task is to determine if a given Web document discusses controversial topics.

However, existing search engines are unlikely to reveal controversial topics to users unless they already know about them [10]. There is an increasing call for search engines to detect these queries and address them appropriately [9, 11]. Previous work [8] presented an algorithm for classifying controversy in Web documents using Wikipedia; this reliance introduces efficiency concerns, and limits the classifier's potential scope to topics covered in Wikipedia. Additionally, to date, little effort has been made to directly model controversy.

In this work, we investigate probabilistic models for automated controversy detection and thus provide theoretic modeling of the problem. First, we develop a probabilistic framework for the controversy detection problem and re-cast the state-of-the-art algorithm from that probabilistic perspective. We use this new perspective to extend prior work from a binary classification model to a probabilistic model that can be used for ranking. We then introduce the *controversy language model*, a new approach to address the controversy detection problem. We develop this language-modeling approach based on a recent qualitative analysis of controversy [5], which is more efficient and easier to compute than the more expensive Wikipedia-based controversy features for automated controversy detection. Finally, we empirically validate our language-modeling approach to automated controversy detection, and find that our approach significantly outperforms the state-of-the-art algorithm for controversy detection [8].

2. RELATED WORK AND BACKGROUND

Controversy detection studies often focus on certain genres or data sources, such as news [3, 13], Twitter [15], and Wikipedia [12, 16, 20]. Since these studies frequently use data-source-specific features such as Wikipedia's edit history features or Twitter's social graph information, existing work cannot be easily generalized to controversy detection on arbitrary webpages.

While some past work uses sentiment as a signal when researching controversy [2, 3], others have argued that opinion and controversy are distinct and non-overlapping concepts [1]. Researchers have shown that using sentiment for controversy detection performs poorly on webpages [8] and that controversy and sentiment are not directly related [13].

Choi et al. attempt to identify controversy and controversial subtopics using various features, particularly a mixture model of topic and sentiment [3]. We depart from their work by directly modeling the probability of controversy.

There also have been a few attempts to detect controversial content with lexicons. Roitman et al. retrieve Wikipedia articles with claims about controversial query topics [17], and Mejova et al. use crowdsourcing to label controversial words [13].

CIKM 2016 Indianapolis, United States

In the social sciences, Cramer [5] explains that "controversy" cannot necessarily be verified to exist in the world independent of its appearance in text, but rather it is created and shaped by the discourse surrounding it, particularly in news outlets. Cramer claims controversy is an "indexical" or "metadiscursive" term, meaning that it can be loosely defined as *something that you would know when you see it* Cramer's work suggests that language could be a key feature in identifying controversy.

2.1 Wikipedia Controversy Features

Previous work studied algorithms that generate scores that signal controversy in Wikipedia [12, 7, 20]. We refer to these scores as Wikipedia Controversy Features (WCF). The scores use various information extracted from Wikipedia pages, meta-data, talk pages, and the page's edit history.

We use the following features to reimplement the kNN-WC algorithm (see §2.2) which relies on them, as well as to collect highly controversial documents for our new language modeling approach:

D Score The presence of the `{dispute}` tag placed on an article by editors. Only ≈200 articles contain this tag.

C Score is generated by a regression-based method [7, 12] that trains on revisions that are labeled with `{controversial}` tags. It uses various metadata features of Wikipedia pages, such as number of unique and anonymous editors.

M Score is generated by Yasseri et al. based on features of edits to predict the ferocity of an "edit war" [20]. Features include the number of mutual reverts of two editors, the number of editors participating in this edit-war, and the editor's reputation.

2.2 kNN-WC Algorithm

Dori-Hacohen and Allan [8] presented an algorithm for identifying whether a given Web document discusses controversial topics. To the best of our knowledge, this work is the first and only attempt to extend the controversy detection problem to general web pages in open domain [8], and built upon past work that investigated controversy detection on Wikipedia. Henceforth, we will refer to their work as k Nearest Neighbors of Wikipedia Controversy (kNN-WC).

The kNN-WC algorithm assumed that the controversy in a web document can be detected from the levels of controversy of related topics. The algorithm modeled topics as related Wikipedia articles, and existing controversy labels on neighbors [12, 20] were used to derive a final judgment on the original document (see §2.1).

Overview of kNN-WC Algorithm: kNN-WC [8] consists of four steps: (1) Find k Wikipedia neighbors given a webpage, (2) compute the three WCF scores (D, C, and M) for each neighbor, (3) aggregate the k values of each WCF score and turn them into three binary labels using thresholds, and (4) vote the labels and make a final decision.

Limitations: kNN-WC is constrained by its dependency on Wikipedia controversy indicators, which are sourced solely from Wikipedia-specific features such as mutual reverts of editors. Search of k-NN for each document is a non-trivial operation, which raises practical efficiency issues. In addition, topical coverage in Wikipedia is necessitated.

In our work, we first contribute a generalization of this algorithm to a probabilistic framework, grounding it in a theoretical conception. We then depart from it by presenting a new language model for controversy.

3. PROBABILISTIC MODELING OF THE kNN-WC ALGORITHM

In this section, we formulate the controversy detection problem with a probabilistic perspective and recast the kNN-WC algorithm as obtaining a probability of controversy for binary classification. We then extend the kNN-WC *algorithm* for binary classification and derive kNN-WC *model*, a probabilistic model that can be used for ranking.

We formulate the controversy detection problem as obtaining the following probabilities: Let D be a document. We define $P(C|D)$ as the probability that D is controversial, and $P(NC|D)$ is the opposite (i.e., non-controversial), and the two probabilities should sum to 1.

$P(C|D)$ can be represented by two components: $P(C, D)$, a joint probability of controversy and D; and $P(D)$, a prior probability of D. For binary classification, we are interested in whether $P(C|D) > P(NC|D)$. However, $P(D)$ is the same for $P(C|D)$ and $P(NC|D)$, and we can thus ignore it in comparison. We model the approach of the kNN-WC algorithm by assuming that $P(C|D)$ can be estimated from the joint probability $P(C, D)$ by the following equation. We then assume that a document and its topics, denoted as T_D are interchangeable.

$$P(C|D) = \frac{P(C, D)}{P(D)} \approx \frac{P(C, T_D)}{P(D)}$$

In the context of kNN-WC, we interpret Wikipedia neighbors as a set of "latent topics" in the document. We denote a set of similar Wikipedia articles to the document (i.e., neighbors) as W_D.

$$P(C, T_D) \approx P(C, W_D)$$

By treating the Wikipedia neighbors as topics, we are able to model $P(C, W_D)$ via WCF (see §2.1). By construction, a WCF reflects some estimate of the co-occurrence of controversy and the topic.

Likewise, kNN-WC algorithm used a binary indicator function $P(C, W_D) \approx c(W_D)$ that outputs 1 (controversial) or 0 (non-controversial). One of the best performing settings of this algorithm [8] is presented below:

$$c(W_D) = \mathbb{1}[\max(WCF_M(W_D)) > \theta_M]$$

where $WCF_M(W_D)$ is a set of M scores for W_D, θ_M is a threshold, and $\mathbb{1}$ converts true to 1 and false to 0.

To support further evaluation of the kNN-WC model, we extend this scoring function to a ranking by removing the threshold. Let $f(W_D)$ be a function that returns an aggregated value of the given neighbors' WCF scores.

$$f(W_D) = \text{agg}(WCF_{\{M,C,D\}}(W_D))$$

We then convert this aggregation function to a probability by normalizing over all possible document neighbors W_D, represented here by a normalization factor Z.

$$P(C|D) \approx \frac{1}{Z} f(W_D)$$

4. CONTROVERSY LANGUAGE MODELS

Cramer [5] manually studies patterns of text surrounding specific terms as `controversy`, `dispute`, `scandal`, and `saga` within the Reuters corpus [18], as being indicative of controversy. Since language may be a good signal, we explore a language modeling approach to this problem. The language

model itself can be derived from WCF features or using Cramer's terms.

Recall that we can say a document is controversial if Eq. 1 is satisfied. If we are only interested in whether $P(C|D) > P(NC|D)$ holds, we can afford rank-safe approximations.

$$\frac{P(C|D)}{P(NC|D)} > 1 \quad (1)$$

Each $P(C|D)$ and $P(NC|D)$ can be represented using Bayes theorem, which allows us to consider the following odds-ratio:

$$\frac{P(C|D)}{P(NC|D)} = \frac{P(D|C)}{P(D|NC)} \cdot \frac{P(C)}{P(NC)} > 1$$

Now our test condition can be expressed as:

$$\frac{P(D|C)}{P(D|NC)} > \frac{P(NC)}{P(C)}$$

where for our purposes, we can treat the right hand side as a constant cutoff (since it is independent of the document D), which can be learned with training data. To avoid underflow, we actually calculate the log of this ratio.

$$\log P(D|C) - \log P(D|NC) > \alpha$$

Therefore, we only have to estimate the probabilities $P(D|C)$ and $P(D|NC)$, which we do using the language modeling framework, construction of a language model of controversy L_C, and a non-controversial language model L_{NC}. We make the standard term independence assumption for each word (w) in our document (D), and avoid zero probabilities with linear smoothing. In practice, we estimate both the general language model [6] and the non-controversial language model as the set of all documents.

$$P(D|C) \approx P(D|L_C) = \prod_{w \in D} (\lambda P(w|L_C) + (1-\lambda) P(w|L_G))$$

$$P(D|NC) \approx P(D|L_{NC}) \approx P(D|L_G) = \prod_{w \in D} P(w|L_G)$$

Here, D_C is a set of controversial documents, and D_{NC} is a set of non-controversial documents, which we estimate in our collections as the background collection, D_{BG}.

$$P(w|L_C) = \frac{\sum_{d \in D_C} tf(w,d)}{\sum_{d \in D_C} |d|}, P(w|L_{NC}) = \frac{\sum_{d \in D_{BG}} tf(w,d)}{\sum_{d \in D_{BG}} |d|}$$

Therefore, to build a language model of controversy, we need to find D_C. We explore WCF features and Cramer-inspired query based models to construct D_C as following:

WCF: Top K Wikipedia articles that have high WCF (e.g, M, C, D) values.

Controversy-indicative terms: Documents that are retrieved by a query believed to indicate controversy. We explore Cramer's terms [5] as well as manual lexicons from past work [13, 17].

5. EVALUATION

We leverage the dataset introduced in prior work [8] that consists of judgments for 303 webpages[1] from ClueWeb09 collection[2]. We perform 5-fold cross-validation and report measures on the reconstructed test set.

[1] `http://ciir.cs.umass.edu/downloads`
[2] `http://lemurproject.org/clueweb09/`

Table 1: Wikipedia-Based Controversy Detection Approaches. All LM approaches have significant improvements over their respective kNN-WC counterpart at the $p < 0.05$ level.

Method	WCF	AUC
kNN-WC model §3	M	0.733
kNN-WC model	C	0.743
kNN-WC model	D	0.500†
LM §4	M	0.801
LM	C	0.835
LM	D	0.795

† In the kNN-WC-D approach, no neighbors were found with dispute tags, so it is equivalent to the weak baseline performance of the *NO* classifier.

We implement our probabilistic model based on kNN-WC, the state-of-the-art approach. Rather than use the full text of the web pages, we follow the kNN-WC algorithm and use the "TF10" query, where the document is approximated by the ten most frequent terms (excluding the 571 SMART stopword list). We experimented with using full pages for classification with our more efficient language-modeling approaches, but results were not statistically different or little worse on average for the LM approaches.

In order to construct D_C, we needed the text of Wikipedia itself. Unfortunately, obtaining the same version of dumps as those used in prior work [7, 8, 20] is nearly impossible. For ease of future reproducibility, we leverage the long abstracts from the 2015-04 release of DBPedia[3].

Prior work reported accuracy; we note that 65% of the 303 documents were non-controversial, so that accuracy does not provide the best view of this dataset. In this work, we primarily present results using the Area Under Curve (AUC) measure, as we can compare performance without tuning thresholds. Since accuracy was used in prior work, we explore accuracy as a measure as well. Compared to kNN-WC algorithm, we improve from 0.72 accuracy (as presented in [8]) and 0.737 accuracy (as reproduced) to 0.779, significant at the $p < 0.001$ level. For our statistical significance tests, we follow in the footsteps of the `pROC`[4], and obtain confidence intervals from bootstrap resamples of the predictions.

For each fold, we trained two parameters by grid search: K, the number of top documents to choose, and λ, the smoothing parameter. For example, to create our M-score-based language model, we ranked the documents in our Wikipedia collection by their M score, and derived a language model based on the concatenation of the top K documents. These models are presented in Table 1.

For building Cramer language models, where the relevant document sets were not created by WCF, but rather by a textual ranking given a query, we used the Galago search engine to rank documents using a query-likelihood retrieval. We explore 6 different corpora as document sources. The K highest-scoring documents were then used as our controversial document set: D_C. These models are presented in Table 2.

6. RESULTS

In Table 1, we present results of our models built around WCF, as introduced in §2.1. All our language modeling approaches are significantly stronger than the k-NN derived

[3] `http://wiki.dbpedia.org/`
[4] `http://cran.r-project.org/web/packages/pROC`

Table 2: Language Models built from documents relevant to Cramer's controversial terms [5]. Collection size $|C|$ in millions of documents and type shown for comparison of results. We found that our wiki dataset was significantly better than all others, which had no pairwise differences otherwise.

Expansion Dataset	Type	$\|C\|$	AUC
DBPedia	Wiki	4.6M	0.853
ClueWeb09B (Spam60)	Web	33.8M	0.741
Reuters	News	0.8M	0.745
NYT-LDC	News	1.8M	0.710
Robust04	News	0.5M	0.711
Signal-1M	News	1M	0.710

Table 3: Language Models built from Cramer's terms and existing lexicons on DBPedia. We find that "controversy" is the most indicative term, and that "saga" is no better than random. Combining terms led to no improvement over "controversy" alone.

Query to build D_C	AUC
`controversy`	0.856
Roitman [17]	0.823
`dispute`	0.740
`scandal`	0.721
Mejova [13]	0.698
`saga`	0.500

approaches. We only report results of WCF features independently because methods of aggregating these features did not improve significantly over the best feature, and these methods were not quite comparable across kNN-WC and LM approaches.

In Table 2, we present an initial exploration of Cramer's hypothesis that news is able to name and define controversy. While Cramer defined four keywords to be indicative of controversy, we find that `controversy` dominates effectiveness on this dataset. We explore these keywords as queries into an expansion corpus, and construct a language model from the highest scoring documents for the given query. That language model is then used for classification.

While we were pleasantly surprised by the efficacy of this simple approach, we did not see the best performance in the news corpora [18] used by Cramer, but rather in using DBPedia as the expansion set. We also explored this approach on other news datasets (Robust04, NYT-LDC [19], and Signal1M [4]) but results were statistically equivalent on all news corpora we tried.

Roitman et al. [17] and Mejova et al. [13] present manually-curated lexicons for controversy tasks. We explore their use intrinsically, with Jaccard Similarity in Table 4 and as queries to build a language model in Table 3.

7. CONCLUSION

We introduce probabilistic approaches to modeling controversy, by both recasting prior work into a theoretical framework, as well as introducing a new model. Using insights from recent social science research, we motivate and explore the first language modeling approach to detecting controversy. We find that our new approach is statistically better than prior work, while simultaneously being more efficient. We demonstrate that strongly indicative terms are as helpful for this problem as complicated Wikipedia-based controversy features and more effective than existing lexicons.

Table 4: A comparison of lexicons built manually and through crowd-sourcing in prior work to our automatically derived language models A (*) indicates significant improvement over the best lexicon approach.

Method	Document	AUC
Roitman Lexicon [17]	TF10	0.543
Mejova Lexicon [13]	TF10	0.562
Mejova Lexicon [13]	Full	0.615
Roitman Lexicon [17]	Full	0.695
Cramer Language Model	Full	0.783
WCF Language Model	Full	0.823*
WCF Language Model	TF10	0.835*
Cramer Language Model	TF10	0.856*

8. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF grant IIS-0910884, and in part by NSF grant IIS-1217281. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor. We thank Peter Cramer and Gonen Dori-Hacohen for thoughtful discussions.

9. REFERENCES

[1] R. Awadallah, M. Ramanath, and G. Weikum. Harmony and dissonance: Organizing the people's voices on political controversies. In *WSDM'12*.

[2] M.-A. Cartright, E. Aktolga, and J. Dalton. Characterizing the subjectivity of topics. In *SIGIR'09*.

[3] Y. Choi, Y. Jung, and S.-H. Myaeng. Identifying controversial issues and their sub-topics in news articles. In *Intelligence and Security Informatics*. Springer, 2010.

[4] D. Corney, D. Albakour, M. Martinez, and S. Moussa. What do a million news articles look like? In *Workshop on Recent Trends in News IR, ECIR'16*.

[5] P. A. Cramer. *Controversy as news discourse*, volume 19. Springer Science & Business Media, 2011.

[6] W. B. Croft, D. Metzler, and T. Strohman. *Search engines: Information retrieval in practice*, volume 283. Addison-Wesley Reading, 2010.

[7] S. Das, A. Lavoie, and M. Magdon-Ismail. Manipulation among the arbiters of collective intelligence: how wikipedia administrators mold public opinion. In *CIKM'13*.

[8] S. Dori-Hacohen and J. Allan. Automated controversy detection on the web. In *ECIR'15*.

[9] S. Dori-Hacohen, E. Yom-Tov, and J. Allan. Navigating Controversy as a Complex Search Task. In *Supporting Complex Search Tasks, at ECIR'15*.

[10] S. L. Gerhart. Do Web search engines suppress controversy? *First Monday*, 9, 2004.

[11] M. Kacimi and J. Gamper. Diversifying search results of controversial queries. In *CIKM'11*.

[12] A. Kittur, B. Suh, B. A. Pendleton, and E. H. Chi. He says, she says: conflict and coordination in wikipedia. In *SIGCHI'07*, pages 453–462.

[13] Y. Mejova, A. X. Zhang, N. Diakopoulos, and C. Castillo. Controversy and sentiment in online news. *arXiv preprint arXiv:1409.8152*, 2014.

[14] E. Pariser. *The Filter Bubble: What the Internet Is Hiding from You*. 2011.

[15] A.-M. Popescu and M. Pennacchiotti. Detecting controversial events from twitter. In *CIKM'10*.

[16] H. S. Rad and D. Barbosa. Identifying controversial articles in wikipedia: A comparative study. In *WikiSym 2012*.

[17] H. Roitman, S. Hummel, E. Rabinovich, B. Sznajder, N. Slonim, and E. Aharoni. On the Retrieval of Wikipedia

Articles Containing Claims on Controversial Topics. In *WWW'16*.

[18] T. Rose, M. Stevenson, and M. Whitehead. The Reuters Corpus Volume 1. In *LREC 2002*.

[19] E. Sandhaus. The New York Times annotated corpus. *LDC*, 6(12):e26752, 2008.

[20] T. Yasseri, R. Sumi, A. Rung, A. Kornai, and J. Kertész. Dynamics of conflicts in wikipedia. *PloS one*, 7(6):e38869, 2012.