Unsupervised Discovery of Opposing Opinion Networks From Forum Discussions

Yue Lu
Twitter Inc.
1355 Market St. Suit 900
San Francisco, CA 94103
yuelu@twitter.com

Hongning Wang, ChengXiang Zhai, Dan Roth
Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61820
{wang296, czhai, danr}@illinois.edu

ABSTRACT

With more and more people freely express opinions as well as actively interact with each other in discussion threads, online forums are becoming a gold mine with rich information about people's opinions and social behaviors. In this paper, we study an interesting new problem of automatically discovering opposing opinion networks of users from forum discussions, which are subset of users who are strongly against each other on some topic. Toward this goal, we propose to use signals from both textual content (e.g., who says what) and social interactions (e.g., who talks to whom) which are both abundant in online forums. We also design an optimization formulation to combine all the signals in an unsupervised way. We created a data set by manually annotating forum data on five controversial topics and our experimental results show that the proposed optimization method outperforms several baselines and existing approaches, demonstrating the power of combining both text analysis and social network analysis in analyzing and generating the opposing opinion networks.

Categories and Subject Descriptors

H.3.m [**Information Storage and Retrieval**]: Miscellaneous; I.2.6 [**Artificial Intelligence**]: Learning

General Terms

Algorithms, Experimentation

Keywords

opinion analysis, social network analysis, optimization, online forums, linear programming

1. INTRODUCTION

Online forum is one of the early applications managing and promoting user generated content. Although being simple in its design – users carry out discussion in the form of message threads, forums remain prevalent and popular even during the recent rise of many sophisticated Web 2.0 applications. As users actively express their opinions and exchange their knowledge on all kinds of topics/issues, e.g., technology, sports, religion, and politics, forums are becoming a great source for opinion mining. However, the simple design of forums combined with rapidly accumulated data make it challenging to make sense out of the forum discussions.

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.

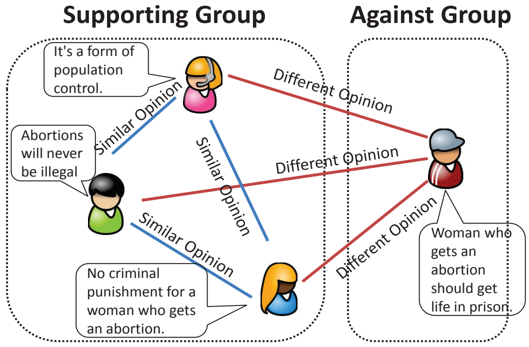

Figure 1: Example Opposing Opinion Network for the Thread on "Abortion"

In this paper, we study an interesting new problem of automatically discovering *opposing opinion networks* from forum discussions, which are defined as latent user groups with strong opposing opinions on different topics, i.e., a supporting group and an against group. There is an example illustration in Figure 1 about opposing user groups on the topic "Abortion". We can see that such discovered opinion networks can serve as a concise and interesting summary of the topics and users in the forum discussions. They can also provide a sense of "virtual community" for the online users, to help them engage in the forum activities more easily. Once we have identified the latent opposing opinion networks, they can enable a number of interesting applications that add social components to forums. For example, we can detect semantically similar topics which involve similar groups of opposing users. We can also find users of similar minds who often agree with each other across different topics or "enemy" users who are often against each other across different topics.

Discovering the opposing opinion networks is related to some existing work on opinion mining, which we will review in detail in Section 6. In short, our work is distinguished from existing work because we exploit the unique characteristics of forum data in an unsupervised way: combining signals from both textual content (e.g. who said what) and social interactions (e.g. who talks to whom). More specifically, from the textual content analysis perspective, we propose two kinds of analysis: (1) topic model analysis of aspect mentions in post text and (2) bootstrapping-based classification of agree and disagree relations between posts. From the social network analysis perspective, we form two assumptions: (1) user consistency across different posts in the same thread and (2) user-user relation consistency in the same thread. Finally, to consolidate all the signals together in a unified way, we design an optimization formulation which can be formulated and solved as a linear programming problem.

There is no existing public data set to evaluate opposing opinion networks, because we need both the textual content and the social iterations among users. So, we created a new data set of five controversial topics in forum discussion, and showed that the proposed optimization method outperforms several baselines and existing approaches, including a SentiWordNet method, a clustering based method and a Max-Cut method .

2. PROBLEM FORMULATION

More formally, a forum F can be considered as a set of discussion threads, i.e. $F = \{TH_1, TH_2, \cdots\}$, where each thread $TH = (T, P, M, R) \in F$ is a tuple of (1) a topic or issue T; (2) a sequence of posts $P = \{d_1, d_2, \cdots, d_n\}$, where each d_i is a post with textual content; (3) a authorship matrix M, where $M_{ji} = 1$ if post d_i was written by u_j; and (4) a partial reply-to relation between posts $R \subset P \times P$, where $R_{ij} = 1$ if d_i replies to d_j.

Assuming that the issue T is given for each thread, which can be obtained from the forum (e.g. "Gay Rights" sub-forum) or can be retrieved using IR methods (e.g. using "Gay Rights" as keywords to retrieve all relevant threads), we aim at automatically discovery of opposing opinion network for each issue:

Definition (Opposing Opinion Network): is a multi-graph (U, E) among forum users $U = \{u_1, u_2, \cdots, u_m\}$ and each edge $(u_i, u_j, t, a^t_{ij}) \in E$ carries an agreement weight $a^t_{ij} \in [-1, 1]$ conditioned on an issue T. And $U = U^+ \cup U^- \cup U^0$ where we are only interested in the supporting/against group U^+ and U^-.

3. METHOD OVERVIEW

It is not trivial to identify each user's opinion for any given issue directly, especially in an unsupervised way, because: (1) treating each user as a single document would lose the rich information at the local post-level (e.g. single post content, reply-to relation); (2) forum users sometimes explicitly express their opinions toward an issue using sentiment words, sometimes not; (3) sometimes users interact and argue with each other, so that expressing their attitudes in an implicit way. Thus, to be most effective in understanding user's opinions, we need to consider the rich information around posts, including users direct opinions toward the issue and indirect attitudes toward other users.

To this end, we propose to decompose the problem: identify each user u_j's opinion o_j in a given thread by aggregating her opinions at the post level, $v(d_i) \in [-1, 1]$ (or v_i for short), i.e, $o_j = \frac{\mathbf{v}^T M_j}{\sum_{i=1}^n M_{ji}}$. Then, given a threshold $t \in [0, 1]$, we can get a supporting user group and an against user group $U^+ = \{u_j | \frac{\mathbf{v}^T M_j}{\sum_{i=1}^n M_{ji}} > t\}$ and $U^- = \{u_j | \frac{\mathbf{v}^T M_j}{\sum_{i=1}^n M_{ji}} < -t\}$. Now, the problem is reduced to identifying opinions in each post, i.e., assigning an opinion score in $[-1, 1]$ to each post d_i as the degree of support or against toward the given issue. This reduction allows us to naturally incorporate a richer set of information when inferring user opinion in each post.

4. IDENTIFY OPINIONS IN POSTS

4.1 Analysis of Textual Content in Posts

We will briefly introduce what a forum post (illustrated in Figure2) is consist of and follow up with more details in the next section.

The Statement Part: A mandatory part of the post, where forum users express their opinions in their own words. Most of previous opinion analysis work only focuses on this part. For example, a representative unsupervised approach is to use SentiWordNet[1] to score the text. Besides the sentiments, the choice of words can be indicative of the opinions too. For example on the "Abortion" issue: a user who posted about "child" and "life" is more likely from an against group, while someone speaking about "rape" and "choice" is more likely from a supporting group.

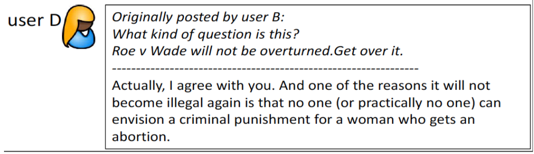

Figure 2: Illustration of a Forum Post

The Quotation Part (i.e., Reply-to): This is an optional part of a post and is usually visualized using different font or color in online forums. In this kind of interaction format, the authors usually directly express their attitude toward the quoted post/user in the first sentence of the reply. It is a very strong indicator the relation between users, if we can automatically classify such sentence as showing users agreement/disagreement, e.g., "totally!" is agreement or "it doesn't make sense to me." shows disagreement. As a result, the quotation part provides us a valuable source to understand the interaction between users.

4.2 Analysis of Agree/Disagree Relations between Posts

We discuss, for each forum thread, how we infer relations between posts as agreement/disagreement, based on multiple signals.

Using "Reply-to" and "User Relation Consistency" With no labeled training data, it is difficult to classify a reply-to text as agreement or disagreement. Because it is impossible to list all the patterns beforehand, i.e., various ways to express users attitude. Thus, we design a bootstrapping method to do classification.

We first extract all the first sentence in quotation text, from the whole forum of more than 1 million posts, and then label these sentences with only a handful of agreement/disagreement patterns (13 in total), such as "I agree" and "I disagree". After that, we bootstrap other patterns with the help of "user relation consistency": suppose we observe one post from u_i replies to a post from u_j and matches the initial "agree" pattern; then we assume that all other reply-to sentences between u_i and u_j in the same thread discussion also follow the "agree" attitude. In this way, we can extract all the "agree" sentences P^{agr} and "disagree" sentences P^{dis} from the whole forum. Essentially, we rely on the users themselves to obtain different ways of expressing agreement or disagreement.

Now, given a new reply-to sentence t_{ij} (indicating that post d_i replies to post d_j), we can just compare the similarity $Sim(t_{ij}, P^{agr})$ v.s. $Sim(t_{ij}, P^{dis})$. Here, $Sim(x, y)$ is the max cosine similarity between a text and a set of text. We can now mark some of the reply-to relations in R as agree $R^{agr} \subseteq R$ or disagree $R^{dis} \subseteq R$, using the following equations:

$$\begin{aligned} R^{agr}_{i,j} &= Sim(t_{ij}, P^{agr}), \quad if \ \frac{Sim(t_{ij}, P^{agr})}{Sim(t_{ij}, P^{dis})} \geq \alpha \\ R^{dis}_{i,j} &= Sim(t_{ij}, P^{dis}), \quad if \ \frac{Sim(t_{ij}, P^{agr})}{Sim(t_{ij}, P^{dis})} \leq \frac{1}{\alpha} \end{aligned}$$

Using "User Consistency": Assuming that most users are logically reasonable, in one single thread, different posts from the same user tend to express consistent opinion. More specifically, suppose

[1] `http://sentiwordnet.isti.cnr.it/`

we know one post from u_i show strong support for a given issue; then all the other posts written by u_i in this thread would be likely to follow this support opinion. Following this assumption, we can encode it as a matrix $A \subseteq P \times P$ from a given thread, indicating agreement relation among posts written by the same author, where $A_{i,j} = 1$ iff d_i and d_j are posts from the same user, i.e.,

$$A_{i,j} = 1, \quad if\ user(d_i) = user(d_j)$$

Using "Framing": It has been found that users with different sentiments/positions would focus on different aspects of the topic, which is called "framing" [18]. For example, on the abortion issue, pro-choice people would emphasize women's rights and freedom while pro-life people would focus on the crude process of abortion. Apparently, these two opposing user groups tend to share similar mentions of aspects within the group and different mentions between groups. To capture this, we first employ a topic modeling method [8] to extract the hidden aspects of discussion, so that we get a number of K aspect models $p(w|\theta)$ for each thread and an aspect distribution $p(\theta|d)$ for each post in this thread. Then, given any two posts from the same thread, if the two corresponding aspect distributions have high positive correlation, their opinions tend to agree; otherwise, their opinions tend to disagree. Denoting $corr(d_i, d_j) = correlation(p(\theta|d_i), p(\theta|d_j))$ as the Pearson correlation coefficients, we can have another measure of post-post relations as agreement $T^{agr} \subseteq P \times P$ or disagreement $T^{dis} \subseteq P \times P$, using the following equations:

$$\begin{aligned} T^{agr}_{i,j} &= \quad corr(d_i, d_j), \quad if\ corr(d_i, d_j) > \beta \\ T^{dis}_{i,j} &= \quad -corr(d_i, d_j), \quad if\ corr(d_i, d_j) < -\beta \end{aligned}$$

4.3 Optimization Formulation

We have introduced and analyzed different signals that can indicate the opinions in forum posts, but it is still not clear how we can combine multiple signals. One way to combine these signals is to use the agree/disagree information as distance measures between posts and then apply clustering-like methods, e.g. MaxCut as in [11]. However, (1) the clustering or partition results cannot tell which group is supporting and which is against; (2) a hard partition does not distinguish users with strong support/against opinions from those with a balanced view. Instead, we propose an optimization formulation that tries to find opinion assignment to each post v_i to capture the different signals introduced before.

Capturing Sentiment Priors: Using the following term, we ensure that our opinion assignment does not deviate too much from the sentiment tagging especially when the sentiment score is high.

$$minimize \sum_{i=1}^{n} |v_i - s_i|$$

Capturing Agreement: With three constructed matrices A, R^{agr} and T^{agr} to encode the signals indicating agreement relation between posts, we are giving a linear penalty if the two opinion scores differ a lot, if we believe the two corresponding posts should agree with each other:

$$minimize \sum_{i=1}^{n} \sum_{j=i+1}^{n} (R^{agr}_{i,j} + T^{agr}_{i,j} + A_{i,j})|v_i - v_j|$$

Capturing Disagreement: We have constructed matrices R^{dis} and T^{dis} to encode the signals indicating disagreement relation between posts. To capture such disagreement, we first separate the representation of "sign" and "absolute value" in each opinion score v_i by introducing two non-negative variables v_i^+, v_i^- and a constraint $v_i = v_i^+ - v_i^-$. In order to ensure that no more than one of v_i^+, v_i^- is positive (the other being zero), we also need to minimize $(v_i^+ + v_i^-)$. In this way: v_i being positive is equivalent to $v_i^+ = v_i$ and $v_i^- = 0$; v_i being negative is equivalent to $v_i^+ = 0$ and $v_i^- = -v_i$; v_i being zero is equivalent to $v_i^+ = v_i^- = 0$.

If there is an entry (i, j) active in R^{dis} or T^{dis}, we want to make the two corresponding opinion scores v_i and v_j have opposite sign but similar absolute value, by the following terms and constraints:

$$minimize \left\{ \sum_{i=1}^{n} \sum_{j=i+1}^{n} [(R^{dis}_{i,j} + T^{dis}_{i,j}) \times (|v_i^+ - v_j^-| + |v_i^- - v_j^+|)] + \mu \sum_{i=1}^{n} (v_i^+ + v_i^-) \right\}$$

subject to $\forall i \in \{1, \cdots, n\}, v_i = v_i^+ - v_i^-$ and $v_i^+,\ v_i^- \geq 0$.

Full Objective Function: Putting every term together, we have the following objective function:

$$\begin{aligned} \mathbf{v} &= argmin_{\mathbf{v}} \left\{ \mu \sum_{i=1}^{n} (v_i^+ + v_i^-) + \lambda_{senti} \sum_{i=1}^{n} |v_i - s_i| \right. \\ &+ \lambda_{agr} \sum_{i=1}^{n} \sum_{j=i+1}^{n} (R^{agr}_{i,j} + T^{agr}_{i,j} + A_{i,j})|v_i - v_j| \\ &+ \left. \lambda_{dis} \sum_{i=1}^{n} \sum_{j=i+1}^{n} (R^{dis}_{i,j} + T^{dis}_{i,j})(|v_i^+ - v_j^-| + |v_i^- - v_j^+|) \right\} \end{aligned}$$

subject to

$$\begin{aligned} \forall i \in \{1, \cdots, n\}, &\quad -1 \leq v_i \leq 1 \\ \forall i \in \{1, \cdots, n\}, &\quad v_i = v_i^+ - v_i^- \\ \forall i \in \{1, \cdots, n\}, &\quad v_i^+,\ v_i^- \geq 0 \end{aligned}$$

where λs and μ are the weights to trade off different components; A, R^{agr}, R^{dis}, T^{agr} and T^{dis} matrices are obtained as described in Section 4.2, while $\mathbf{v}$, $\mathbf{v}^+$ and $\mathbf{v}^-$ are the variables. With this formulation, we can use standard techniques to transform it to a Linear Programming problem and solve it efficiently.

Topics	# Posts/User	# Posts/Thread	# ReplyTo
abortion	3.19	59.4	27
healthcare reform	3.85	64.6	29.2
illegal immigrants	2.94	61.4	24.6
iraq war	3.31	64.8	26.4
president obama	3.22	61.8	24.8

Table 1: Basic Statistics of Data Sets

5. EXPERIMENTS

5.1 Data sets and Human Annotation

We created our own data sets from an online military forum[2]. We crawled 43,483 threads of discussions, containing 1,343,427 posts, from "Hot Topics & Current Events". In order to make it easier for the human judges to annotate, we further narrowed down to five popular and controversial topics , and applied information retrieval method to retrieve the top five most relevant threads for each topic. The basic statistics are in Table 1.

We first asked our colleagues to label each post as "Support", "Against" or "Not Sure" about the given topic. In total, we have collected 230 posts with agreed labels, where 26% as "Support", 41% as "Against" and 33% as "Not Sure". In particular, the true disagreement rate among the judgments (i.e., "Support" v.s. "Against") is low: only 12.31%, showing that the task is designed reasonably. Then, we further utilized the crowd sourcing service through

[2] `forums.military.com`

CrowdFlower[3] to get more labels on all 1584 posts. For better quality control, we further required each post to be labeled by at least three annotators. The annotation results basically followed the statistics of the first round of annotation: 30% as "Support", 43% as "Against" and 26% as "Not Sure". Also, according to CrowdFlower's statistic, the agreement among the annotators is 0.7584. We will use the CrowdFlower annotation[4] as our ground truth.

5.2 Methods for Comparison

LP: Our proposed method, solved using PyGLPK toolkit[5]. All experiments are performed when setting K the number of topics to be 5, and the threshold α and β to 2.

UserClustering: We build similarity graph among users from cosine similarity of their bag-of-words representation and apply graph partition based clustering of two groups of users (using CLUTO[6]).

SentiWordNet: We tag each post by taking the average the SentiWordNet opinion score of words, then each user's opinion is the averaged opinion score of all her posts. It represents an unsupervised sentiment analysis method which only relies on the text.

MaxCut: The method proposed in [11] is solving a similar problem. They first classify each reply-to relation as agree/disagree/neutral using a pattern dictionary; then user-user relation is defined as a linear combination of their post-post relations (positive if disagree and negative if agree); finally, a MaxCut algorithm is performed on this user-user graph to generate the user groups partition. Since we do not have their algorithm implementation or their pattern dictionary, we use SentiWordNet and the same set of patterns from our method as the first step classifier. Then we use a semi-definite non-convex programming solver [7] to sovle the MaxCut problem.

5.3 Evaluation of Agree/Disagree Classification

We first evaluate the accuracy of the local classification of relations between posts. We derive the ground truth of post-post relations from post level judgment, and ignore the cases when either post is annotated as "Not Sure". Note that, instead of aim at inferring the relations between *every* pair of posts, we only need a subset of high-confidence pair-wise relations, because we will only use them to better predict the point-wise labels.

In Table 2, we compare the results[8] from the first step of MaxCut and our methods of extracting matrices R (derived from reply-to), T (derived from topic modeling), and A (derived from user consistency assumption). Top part of the table shows that our method outperform the MaxCut method in all metrics. In particular, the rule-based classifier in MaxCut cannot handle different vocabulary and possibly slangs in the forums, resulting in very low recall. Our classifier is learned by exploiting the forum data itself, including both textual analysis and social network analysis.

To further understand each signal in our method, we evaluate them individually in the bottom part of the table. (1) The A matrix

Method	Precision	Recall	F1 Measure
MaxCut	0.4732	0.0090	0.0177
$R+T+A$	0.6010	**0.1942**	**0.2936**
R	0.5582	0.0036	0.0071
T	0.5632	0.1134	0.1888
A	**0.6791**	0.0900	0.1589

Table 2: Accuracy of Agree/Disagree Classification

performs the best in precision, indicating that author consistency assumption is most accurate in identifying post-post relation. (2) The R matrix is sparse as indicated by the lowest recall, because it relies on explicit reply-to relations, which is only a small subset as in the forum data. In comparison, A matrix relies on multiple posts written by the same user, which is often the case, thus providing higher recall. The T matrix only depends on the post content, thus applies to all posts, generating the highest recall. (3) Finally, the recall of $R+T+A$ is almost the same as the sum of that of the three matrices, suggesting that these three matrices provide complementary information. Thus, by combining them together, we can get the most comprehensive information about post-post agree/disagree.

Now that we have shown that our methods discover more accurate relations *between posts*, in the next set of evaluation we will further test the performance of using these post-post relations to discover opposing opinion network.

5.4 Evaluation of Opposing Opinion Network

Since we only care about the strongly opposing users, we only take as ground truth: users with at least two posts and aggregated ground-truth opinion score $|s| > 0.5$. This results in 57 supporting users and 78 against users for the selected 5 topics. Note that the ground truth is only a subset of users that we have confident human labels, so we can only evaluate the Accuracy on this subset. We also evaluate Mean Squared Error (MSE).

In the top half of Table 3, we compare three baselines with our Linear Programming (LP) method. We use a threshold $t = 0$ to decide the supporting/against group for SentiWordNet and LP. We can see that

- UserClustering is not performing well, which shows that treating each user as a big document and ignoring the relations among posts is not effective.
- MaxCut (the original implementation as in [11]) is not performing well either, suggesting that only considering agree/disagree relations among users is not effective enough. In particular, the support and against groups are very unbalanced, which is because of the sensitivity of partition based methods.
- SentiWordNet (t=0) provides better accuracy than the previous two baselines, suggesting that the sentiment score of the text the users posted is a reasonable way to infer their opinion group.
- Our LP (t=0) method clearly outperforms all three baselines in all measures. This is because we include both "who said what" (through the sentiment term) and "who talks to whom" (through the agree/disagree terms) in the objective function to understand the user's opinions in a more comprehensive way. Since we use SentiWordNet as one term in the objective function, this shows that the other terms capturing agreement/disagreement further help adjusting the opinion scores more accurately.

We further conduct more analysis experiments in order to (1) allow more fair comparison with baselines, and (2) better understand the performance gain of our LP method over existing baselines. The results are summarized in the second half of Table 3.

- MaxCut+: We first try to improve MaxCut by using our full set of heuristics as input in addition to the sentiment heuristic in the

[3] www.crowdflower.com
[4] data and annotation available at http://sifaka.cs.uiuc.edu/~yuelu2/forumdata/
[5] http://tfinley.net/software/pyglpk/
[6] http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview
[7] http://www.stanford.edu/~yyye/Col.html
[8] Note that the absolute numbers of recall (thus also F1 measure) in are generally very low, but this does not mean the algorithms are performing poorly. Since we created some synthetic agree/disagree pairs to test the classification results, N point-wise labels result in $N * N$ pair-wise ground truth of agree/disagree samples, which is a large number for the denominator of recall calculation (most of them we will not encounter in real data).

Method	Accuracy (For)	Accuracy (Against)	Accuracy (For+Against)	MSE
UserClustering	0.6250	0.4615	0.5299	0.4535
MaxCut (original)	0.6071	0.4487	0.5149	0.4528
SentiWordNet (t=0)	0.6250	0.5250	0.5670	0.4470
LP (t=0)	**0.6429**	**0.5513**	**0.5896**	**0.4103**
MaxCut+	0.0714	0.8590	0.5299	0.4444
LP*	0.5893	0.5513	0.5672	0.4149
SentiWordNet*	0.6964	0.4103	0.5299	0.4631

Table 3: Accuracy of User Opinion Prediction

original paper. However, since we introduce many agreement heuristics which get translated into negative edge weights, MaxCut will produce even more unbalanced output, so as to maximize the weight sum of cut edges. It is clear that MaxCut (with full heuristics) shows large difference between the accuracy in two groups. It suggests that the improvement gain of our LP method comes from not only the useful heuristics but also the effective optimization formulation.

- LP* and SentiWordNet*: In order to account for the possible bias of unbalanced partition in MaxCut, we further evaluate SentiWordNet and LP using the same partition size as MaxCut. More specifically, we first rank the users by the opinion scores output by SentiWordNet or LP, and then partition into user groups is done so that the ratio of supporting users v.s. against users is the same as that from MaxCut output. As can be seen in the table, both methods still outperform MaxCut, showing that the output opinion scores are not only more flexible than a hard partition but also more accurate.

In addition, we tried varying the weights in the objective function 4.3, and improved prediction accuracy over other methods are consistent as long as we leave sufficient weight on sentiment scores λ_{senti} (i.e., at least twice as much as λ_{agr} and λ_{dis}) while maintaining a small μ, e.g., 0.01. We will leave automatic weight learning as future work.

6. RELATED WORK

Discovering opposing user networks is a new problem different from existing work, but there are also important connections. First, opinion analysis in text data has received extensive attention these years [10, 4, 13, 9, 5, 12, 7], but these methods focus on predicting a sentiment class/rating of the text *individually*, so we cannot take the rich social interaction context into consideration. In the experiments, we included a baseline using SentiWordNet that represents an unsupervised document level sentiment analysis work.

Related work (e.g. [17, 2, 16, 19, 3]) on identifying *user* opinions (or stands, views) is usually defined as a classification problem of the users into predefined groups. Instead, we are only interested in a subset of users who are strongly against each other. Nevertheless, this line of related work is highly relevant. However, these previous work either use the textual content without much consideration of social interactions among users (e.g. [6, 15, 14]) or use the social link information without using much of the content information (e.g. [1, 11]). Instead, our work has shown that the combination of content analysis and social network analysis is most powerful for inferring user opinions.

Some of the work (e.g., [16, 2]) have also explored the relations among users. In comparison to those supervised methods, we study the user opinions in the more difficult unsupervised setting, and we propose an effective optimization formulation to combine signals from unsupervised content analysis and social network analysis. Moreover, our unsupervised approach can be easily incorporated into any supervised learning as a way to provide intelligent features.

7. CONCLUSIONS

In this paper, we study an interesting new problem of discovering opposing opinion networks from forum discussions, which are essentially a type of latent social networks based on user opinions: the strongly opposing user groups. We propose a method to discover them in an unsupervised way using signals from both textual content and social interactions. We also created a manually annotated forum data set on five controversial topics, and demonstrated that the proposed optimization method outperforms several baselines and existing approaches.

Our work is the first step in a novel text mining direction where the focus is on analyzing latent user behavior and social structure behind text. The opposing opinion network we studied is only a simple form of general opinion networks, which may include multiple user opinion groups. For example, we may further infer more specific relations among users from the opinion network, e.g. friends, enemies, followers, etc.

8. REFERENCES

[1] R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu. Mining newsgroups using networks arising from social behavior. In *WWW '03*, pages 529–535. ACM.

[2] M. Bansal, C. Cardie, and L. Lee. The power of negative thinking: Exploiting label disagreement in the min-cut classification framework. *COLING'08: Companion volume: Posters*, pages 13–16.

[3] C. Burfoot, S. Bird, and T. Baldwin. Collective classification of congressional floor-debate transcripts. In *ACL'11: Human Language Technologies*, pages 1506–1515.

[4] H. Cui, V. Mittal, and M. Datar. Comparative experiments on sentiment classification for online product reviews. In *Proceedings of the National Conference on Artificial Intelligence*, page 1265, 2006.

[5] K. Dave, S. Lawrence, and D. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *WWW'03*, pages 519–528. ACM.

[6] M. Galley, K. McKeown, J. Hirschberg, and E. Shriberg. Identifying agreement and disagreement in conversational speech: use of bayesian networks to model pragmatic dependencies. In *ACL '04*.

[7] A. Goldberg and X. Zhu. Seeing stars when there aren't many stars: graph-based semi-supervised learning for sentiment categorization. In *TextGraphs-1*, pages 45–52. ACL, 2006.

[8] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR'99*.

[9] S. Kim and E. Hovy. Determining the sentiment of opinions. In *COLLING'04*, pages 1367–1373. ACL, 2004.

[10] B. Liu. Sentiment analysis and subjectivity. *Handbook of Natural Language Processing Second Edition*, pages 1–38, 2010.

[11] A. Murakami and R. Raymond. Support or oppose?: classifying positions in online debates from reply activities and opinion expressions. In *COLING '10: Posters*, pages 869–875.

[12] B. Pang and L. Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL'05*, pages 115–124. Association for Computational Linguistics.

[13] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *ACL'02*, pages 79–86. Association for Computational Linguistics, 2002.

[14] S. Park, K. S. Lee, and J. Song. Contrasting opposing views of news articles on contentious issues. In *ACL'11*, pages 340–349, 2011.

[15] S. Somasundaran and J. Wiebe. Recognizing stances in online debates. In *ACL/AFNLP'09*, pages 226–234.

[16] S. Somasundaran and J. Wiebe. Recognizing stances in ideological on-line debates. In *CAAGET'10, NAACL-HLT'10*, pages 116–124. Association for Computational Linguistics, 2010.

[17] M. Thomas, B. Pang, and L. Lee. Get out the vote: Determining support or opposition from Congressional floor-debate transcripts. In *Proceedings of EMNLP*, pages 327–335, 2006.

[18] D. Tversky, Amos; Kahneman. The framing of decisions and the psychology of choice. pages 453–458, 1981.

[19] A. Yessenalina, Y. Yue, and C. Cardie. Multi-level structured models for document-level sentiment classification. In *EMNLP'10*, pages 1046–1056. Association for Computational Linguistics.