2015

Mining user viewpoints in online discussions

Minghui QIU
Singapore Management University, minghui.qiu.2010@phdis.smu.edu.sg

Mining User Viewpoints in Online Discussions

by

Minghui Qiu

Submitted to School of Information Systems in partial fulfillment of the requirements for the Degree of Doctor of Philosophy in Information Systems

Dissertation Committee:

Jing JIANG (Supervisor / Chair)
Assistant Professor of Information Systems
Singapore Management University

Feida ZHU
Assistant Professor of Information Systems
Singapore Management University

David LO
Assistant Professor of Information Systems
Singapore Management University

Aixin SUN
Associate Professor with School of Computer Engineering
Nanyang Technological University

Singapore Management University
2015

Mining User Viewpoints in Online Discussions

Minghui Qiu

Abstract

Online discussion forums are a type of social media which contains rich user-contributed facts, opinions, and user interactions on diverse topics. The large volume of opinionated data generated in online discussions provides an ideal testbed for user opinion mining. In particular, mining user opinions on social and political issues from online discussions is useful not only to government organizations and companies but also to social and political scientists. In this dissertation, we propose to study the task of *mining user viewpoints or stances* from online discussions on social and political issues. Specifically, we will talk about our proposed approaches for these sub-tasks, namely, viewpoint discovery, micro-level and macro-level stance prediction, and user viewpoint summarization.

We first study how to model user posting behaviors for viewpoint discovery. We have two models for modeling user posting behaviors. Our first model takes three important characteristics of online discussions into consideration: user consistency, topic preference, and user interactions. Our second model focuses on mining interaction features from structured debate posts, and studies how to incorporate such features for viewpoint discovery. Second, we study how to model user opinions for viewpoint discovery. To model user opinions, we leverage the advances in sentiment analysis to extract users opinions in their arguments. Nevertheless, user opinions are sparse in social media and therefore we propose to apply collaborative filtering through matrix factorization to generalize the extracted opinions. Furthermore, we study micro-level and macro-level stance prediction. We propose an integrated model that jointly models arguments, stances, and attributes. Last but not least, we seek to summarize the viewpoints by finding representative posts as one may find the amount of posts holding the same viewpoint is still large.

In summary, this dissertation discusses a number of key problems in mining user viewpoints in online discussions and proposes appropriate solutions to these problems. We also discuss other related tasks and point out some future work.

Table of Contents

List of Figures

List of Tables

To Cen, my parents, and in-laws.

Acknowledgments

This thesis would not have been possible without support from many people.

First and foremost, I would like to thank my advisor and role model, Assistant Professor Jing Jiang, for many helpful and insightful technical conversations during my PhD study. Jing has provided me with the right amount of freedom to explore various research problems and given me guidance through out my PhD years. Her valuable encouragement, guidance, and comments have helped to shape my interests and ideas, which is of a great support for my dissertation. I thank her also for reading my often last-minute paper drafts, helping to correct my countless grammar errors, and giving me useful feedback. I have been really impressed with how dedicated she is to her research and to her students; and mere thanks likewise are not enough to express my gratitude to her.

I am very thankful to my dissertation committee comprising of Assistant Professor Feida Zhu, Assistant Professor David Lo, and Associate Professor Aixin Sun for their research vision and commitment. They have been supportive beyond the call of duty, and have provided constructive comments and suggestions that hold this work to a high standard (that said, any errors in this dissertation are my own). I have collaborated with Feida on some papers and for which I am extremely indebted to his invaluable comments for shaping a research idea, presenting my research, and developing suitable math models.

I would like to thank Professor Ee-peng Lim and Assistant Professor Hady Lauw for their great encouragement and insightful feedback on my research. I extend my thanks to Professor Steve Fienberg and Ee-peng Lim for giving me the opportunity

to participate in PhD Overseas Training Residency in CMU. I would also like to thank Associate Professor Noah Smith, Professor Alex Smola, and Chong Wang for their great support and valuable guidance during my exchange in CMU. The stance prediction work in Chapter 5 have benefited from guidance by Noah and helpful discussions with Yanchuan Sim. I have obtained valuable experience on large scale machine learning during my internship in Google. For this, I would like to thank my mentors and collaborators at Google Strategic Technologies Team. Especially, I want to thank Dr. Vanja Josifovski, Amr Ahmed, Yuan Wang, and Alex Smola who offered me critical suggestions and insightful perspectives on my projects.

I also want to thank my many amazing co-authors (in chronological order), Yaliang Li, Liu Yang, Swapna Gottipati, Wei Dong, Qiming Diao, Chao-Yuan Wu, Lizi Liao, Fenglong Ma, Yinfei Yang, Forrest Sheng Bao, Yang Li and Jing Gao. I am lucky enough to work with so many great people, and without their supportive comments and suggestions, I may not be able to complete this dissertation. I really enjoyed collaborating with you. It is not only my co-authors who helped making my PhD years more fun and productive. I would also like to thank my amazing friends in SMU, Ming Gao, Ying Ding, Wei Xie, Shaowei, Juan Du, Wei Gong, Jiali, Su Mon, Runquan, Felicia, Na Fu, Hanbo, Yazhe, Larry, and Tuan-Anh, with whom I shared my precious graduate life. I have been benefited a lot from you through many brainstorming sessions, discussions, and research collaborations.

Finally, I would like to express my deep thanks to my family and friends for their moral and emotional support. I would not be able to make it without you. Many thanks especially to my dear Cen, brother Zaihui, Linfang, sister Zaiyan, Xianbo, my parents, and in-laws for always being with me, and for their unconditional love. I dedicate my dissertation to all of you.

Chapter 1

Introduction

Online social media such as social networks, blogs, forums and debates provides ample opportunities for netizens to express their opinions. The large volume of opinionated data generated in online social media is an ideal testbed for user opinion mining. In particular, mining user opinions on social and political issues is useful not only to researchers but also to government organizations and companies. In this dissertation, we study the task of *mining user viewpoints* on social and political issues in online social media.

User *viewpoint* or *stance*[1] refers to an overall position held by a person toward an object, idea or proposition [92]. For example, a person may either support or oppose the "Affordable Care Act." To discover or predict user viewpoints from social media is central to understanding public opinion towards a controversial issue. Particularly, to discover or predict user viewpoints toward issues in domains such as politics and religion is useful to policy makers and government organizations. It supports a wide range of applications, including identification of social groups [83], ideological groups [2], user demographic information [31], and building better recommender or personalization systems [66].

With the popularity of online social media and advances of search engines, for a heated or controversial issue, one may easily find a large number of user comments,

[1] We use the terms *viewpoint* and *stance* interchangeably in this dissertation.

posts, or web pages on it. For example, for the threat of "Ebola", we can find more than 30 million search results in a commercial search engine such as Google. For a policy maker or domain expert who wants to know the public opinions on how to defeat the threat of Ebola, the first question he may ask is what the major viewpoints are on the threat of Ebola. In this case, there is a need to discover the major viewpoints from the large number of user-generated data on the threat of Ebola. We refer to this task as *viewpoint discovery*, whose main goal is to find the major viewpoints on a given issue. For the above example, the major viewpoints probably include "we should ban people from traveling to west Africa", "we should send more health workers to west Africa", "we need to work harder to find vaccination for Ebola", etc. Given the major viewpoints on the issue, an immediate next question may be what the percentage of users holding a certain stance is for the issue. We refer to this task as *macro-level stance prediction*. However, to do this task, one may first need to predict a particular user's viewpoint on the issue, i.e., micro-level stance prediction. An aggregation of all users' viewpoints on the issue provides macro-level stances. Now, given the discovered major viewpoints and corresponding posts or users holding those viewpoints, one may find the number of posts or users holding a certain viewpoint is still large. For example, we may find a large number of posts or users taking the viewpoint of "we need to work harder to find vaccination for Ebola". A plausible way to help understand these viewpoints is to extract the essence of all the related information through *viewpoint summarization*. Here we define viewpoint summarization as a task to find the representative posts for each viewpoint of a given issue.

In summary, mining user viewpoints on an issue includes these four tasks.

- *Viewpoint discovery.* Given an issue and all the comments/posts on this issue, this task is to cluster posts or users into different viewpoint groups. In this case, the viewpoints may not be simply two-sided, i.e., to support or oppose something or someone, but rather they can be different perspectives on an issue.

- *Micro-level user stance prediction.* This task is to predict a given user's stance on a target issue on which there is no metadata such as vote-up and vote-down to explicitly indicate the user's stance. The challenge here is cold-starting, i.e., some users may not have any past stances on related issues or arguments on the target issue. In cases where user may have expressed his or her stance in text, we have to draw clues from the arguments on the respective issue. In cases where user has not commented on an issue, we may also be able to infer her stance based on her past stances on related issues.

- *Macro-level user stance prediction.* We also consider macro-level stance prediction, where we estimate the percentage of users holding a certain stance for a particular issue. This can be done by aggregating micro-level stance prediction results on all users or a group of representative users.

- *Viewpoint summarization.* Given a controversial issue and all the comments/-posts grouped by their viewpoints, our task is to find representative posts for each viewpoint.

1.1 Motivation

Traditional means of user viewpoint discovery include polls and surveys. While they have shown to be very effective, for example the opinion polling for the U.S. presidential election[2], they require a large number of manpower support.

With the rise of online social media, there is a growing interest in inferring public opinion from freely available online social media texts and metadata [68, 72, 95]. Such approaches have the potential to *complement traditional surveys and polls*. Among the various social media platforms, online discussions are a type of social media which contains rich user-contributed texts, opinions, and interactions on di-

[2]http://en.wikipedia.org/wiki/Historical_polling_for_U.S._Presidential_elections

verse topics. Online discussion platforms range from online forums[3] and debates[4] to any websites supporting "comment" or allowing a formation of discussions. In fact, in the era of web 2.0, all the major websites allow users to comment and form discussions. Examples include Wikipedia, Facebook, Twitter, Google+. They provide a rich wealth of information for the task of user viewpoint discovery. In a nutshell, the advantages of mining user viewpoints form such online discussions include the following: 1) Online discussion platforms allow netizens to express their opinions, to ask for advice, and to form online communities; they usually have a large user base which may be representative for the public. For example, the most popular online discussion forum in China, "Tianya Club", has over 94 million of registered users[5]; 2) Responses to major sociopolitical events and issues can be found in discussion forums. For example, after the presidential debate between Barack Obama and Mitt Romney, there were heated discussions in online forums such as CreateDebate[6].

In this dissertation, we study the task of mining user viewpoints in online discussions. We will focus on four tasks, namely, viewpoint discovery, micro-level stance prediction, macro-level stance prediction and user viewpoint summarization. In a netshell, we visualize the task of mining user viewpoints in Figure 1.1.

1.2 Challenges

Online discussion forums have been explored in the past for predicting user stances [92, 93, 74]. However, there are data structures and properties of online discussions which we need to be aware of when we design user viewpoint mining methods. Specifically, we will consider the following important characteristics of online discussion data for mining user viewpoints.

Threaded-structure. Arguments are organized in threaded structures. Each argu-

[3] www.tianya.cn, forums.asiaone.com/

[4] wiki.idebate.org,www.createdebate.com, www.debate.org

[5] www.tianya.cn

[6] www.createdebate.com

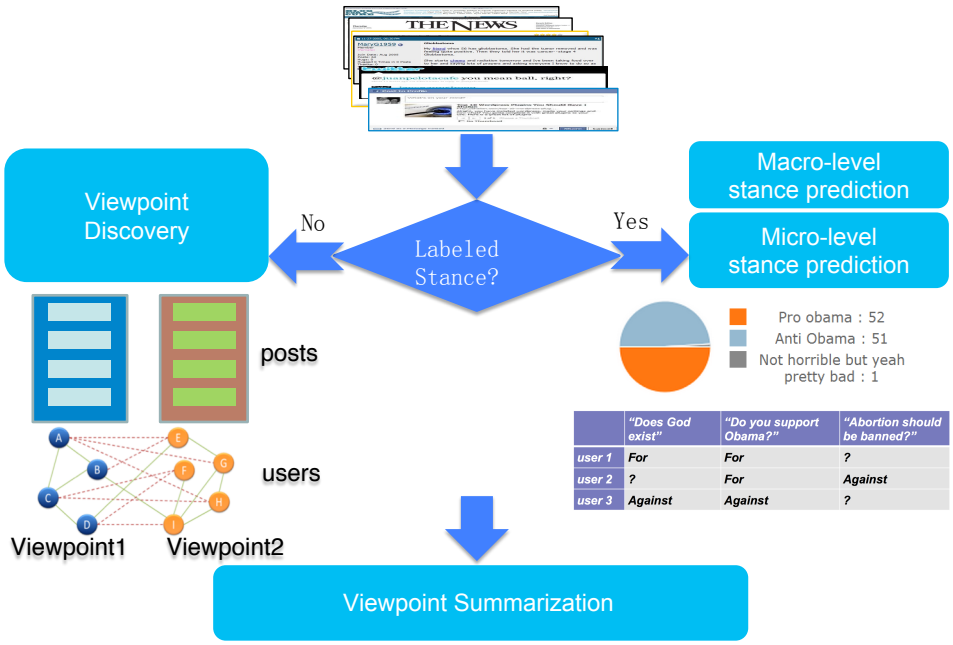

Figure 1.1: An overview of the task of mining user viewpoints.

ment can be an independent post or a reply to an earlier argument. To differentiate between these two types of posts is useful, as the former tends to contain more independent contents, while the latter contains more agreement or disagreement expressions.

User interactions and opinions. A thread is like a conversation, where users not only directly comment on the issue but also comment on each other's posts. The interaction expressions in the exchanged posts may help to infer the relation between two users and subsequently infer the viewpoints of the corresponding posts. Furthermore, to back up their viewpoints, users may also express their opinions toward other opinion targets besides the recipient users. To mine such opinions can further help the task.

Low user participation rate. There may be a low online participation rate of Internet users in online discussion forums relating to any particular issue. To infer a user's viewpoint on the issue, we need to draw on clues from other related issues on which a user's viewpoint has been explicitly expressed as well as other users with similar patterns of viewpoints.

Rich user attributes. Users may also reveal their personal information. In fact,

major discussion forums allow users to reveal their demographic information. For example, CreateDebate website allows users to state their age, nationality, political orientation, religion, etc. How to make use of such information for the task of user viewpoint discovery is interesting to explore.

In brief, to mine user viewpoints from online discussions involves a fine grained analysis of issues, users, arguments, interactions and opinions in the context of threaded structure.

1.3 Objective and contribution

In this dissertation, our research objective is to propose principled approaches for these three tasks in online discussions, namely, viewpoint discovery, micro-level and macro-level stance prediction, and provide some empirical studies on viewpoint summarization. We propose probabilistic models for mining user viewpoints by taking advantage of the availability of different sources of information in online discussions.

Below we give a high-level overview of the various chapters of this dissertation.

- *Modeling user posting behaviors for viewpoint discovery.* In this part, we study how to model each user's posting behaviors in terms of how he chooses topics and interacts with others when he expresses his viewpoint in online discussions. We will discuss our two works related to this task.

  Our first model takes three important characteristics into consideration, namely, user consistency, topic preference, and user interactions. The first one refers to the observation that a user's opinion on an issue usually remains unchanged during a certain time period. The second one models the observation that users with different viewpoints tend to focus on different topics. This is close to a phenomenon called "framing" [96]. The last one refers to the observation that in a forum thread, like in conversations, users interact with each other by commenting on each other's posts. Thus, modeling

the agreement/disagreement expressions among users can help find different viewpoints.

As shown in above work, to tackle the task, it is important to exploit user posts that implicitly contain support and dispute (interaction) information. The challenge we face is how to mine such interaction information from the content of posts and how to use them to help identify stances. We propose a two-stage solution based on latent variable models: an interaction feature identification stage to mine interaction features from structured debate posts with known sides and reply intentions; and a clustering stage to incorporate interaction features and model the interplay between interactions and sides for debate side clustering. Empirical evaluation shows that the learned interaction features provide good insights into user interactions and that with these features our debate side model shows significant improvement over other baseline methods.

- *Modeling user opinion matrices for viewpoint discovery.* Our third model addresses the sparsity of user interactions in online discussions. We first make use of the advances in sentiment analysis to extract user opinions in online user interactions. Based on this, we build two user opinion matrices, one for user interactions, the other for user-aspect opinions. As these user opinion matrices are still very sparse, we propose to apply collaborative filtering through matrix factorization to generalize and improve the extracted opinion matrices from forum posts. The resulting low-rank latent factor representations of users makes it feasible to cluster users by their viewpoints.

- *Micro-level and macro-level stance prediction.* We study viewpoint discovery for a new issue which has a low online participation rate of Internet users in online discussion forums. We propose an integrated model that jointly models texts, user viewpoints and social networks. We consider hidden factor models to model user viewpoints, and user texts to give a human-interpretable

explanation for each hidden factor. We also incorporate social context to boost the model performance. Our experiments show promising results on both micro-level stance prediction and macro-level stance prediction.

Last but not least, we have an empirical study on user viewpoint summarization by leveraging information learnt by our viewpoint discovery model. We acknowledge that these studies are no way near the exhaustive list of tasks in user viewpoint mining, but rather a set of important tasks that need to be considered beforehand. We leave the discussion of other related tasks to Chapter 7.

1.4 Dissertation structure

The studies presented in this dissertation were originally reported in [80, 82, 83, 84]. The dissertation gives a more thorough extensions to them and present more detailed results. The reminder of this dissertation is organized as follows: Chapter 2 is a literature review which examines closely related research work. Chapter 3 investigates how to model user posting behaviors for viewpoint discovery. Chapter 4 studies how to extract user opinions from texts and how to use them to help the task of viewpoint discovery. Furthermore, Chapter 5 proposes an integrated model to jointly model user arguments, interactions, and attributes. Chapter 6 proposes to summarize viewpoints by leveraging information leant by our viewpoint discovery model. Finally, Chapter 7 summarizes the contributions of this dissertation.

Chapter 2

Literature Review

In this chapter, we discuss related work to the task of mining user viewpoints in online discussions. We will introduce four types of work, namely, viewpoint discovery, micro-level stance prediction, macro-level stance prediction, and user viewpoint summarization.

2.1 Viewpoint discovery

Existing approaches for viewpoint discovery have focused on these types of information mined from text, (*i.*) Users arguments directly related to the issue [18, 53, 73], i.e. those users' arguments that are related to statements to support their main claim. For example, for someone who supports gun control, his/her arguments may include "gun control can reduce the number of murders every year," "gun control reduces suicide," etc. Here each statement can be seen as a reason to support the main claim. (*ii.*) User interactions, which indicate whether two users hold the same viewpoint or not [1, 19, 11, 7, 84]. Users also write arguments to reply to other users. For users with different viewpoints, we may see many arguments with disagreement. And for users with the same viewpoint, we may find more agreements. For example, we find this post between two users with different viewpoints: "Actually, I have to disagree with you." (*iii.*) Users opinions towards certain entities closely related

to the issue, which are often correlated with the users stance on the issue [2, 83]. Users also express their opinions towards some issue related entities, for example, many users pose their opinions on "Republican Party" when discussing the issue on "Do you support Obama?" These opinions are often correlated with user's stance. For example, we find a "Support-Obama" user posted his disappointment towards Republican party: "I simply point out how absolutely terrible the Republican party is." Below we discuss how these types of information can be used for viewpoint discovery.

2.1.1 Arguments related to the issue

The availability of texts associated with issues and user viewpoints provided a rich source of linguistic data from which we can improve viewpoint discovery. Assuming users' arguments are mostly related to the issue, one may find that users with different viewpoints tend to focus on different topics. For example, for someone who supports gun control, his arguments may be more on the negative sides of using guns like "murders" or "suicide" involved with guns. While for someone who opposes gun control, his arguments may be on: citizen gun ownership acts as a deterrent against criminals, or education on safe gun ownership can reduce the gun risks. Thus many existing studies [18, 53, 73] make use of viewpoint-specific topic preference to help predict user viewpoints, as discussed in the following.

Paul et al. [73] used a topic-aspect model to jointly model topics and viewpoints. They assume these two concepts are orthogonal. The work assumes that each user has written an article about his/her viewpoint on an issue. While there are multiple topics associated with the given issue, they assume that users with different viewpoints will use different words when talking about the same topic. Fang et al. [18] proposed a model that also combines topics and viewpoints. They assume that documents are already grouped by viewpoints, based on which they extract viewpoint-specific word distributions. The work in [92] mined the web to augment existing

data with learnt associations that are indicative of opinion stances in debates. In our work [80], we study each user's posting behaviors in terms of how he chooses topics and interacts with others when he expresses his viewpoint in online discussions. The model takes three important characteristics into consideration, namely, user consistency, topic preference, and user interactions. One important factor of this paper is that we model two types of textual content, one for inferring topics, the other for inferring user interaction polarity. Experiments show it is beneficial to extract texts related to user interactions and model the interplay between user interactions and viewpoints. This work is more suitable for online discussion posts as rich user interactions are observed.

2.1.2 User interactions

Besides users' arguments related to the issue, we also observe a large amount of posts in online discussions are related to user interactions. A user interaction refers to texts exchanged between users that indicate whether two users hold the same viewpoints or not. For example, this post is a user interaction between two users with different viewpoints: "Actually, I have to disagree with you."

To infer user interaction polarity is crucial to viewpoint discovery as users within the same viewpoint tend to have positive interactions while users with different viewpoints tend to have negative interactions. To infer interaction polarity is related to detecting agreement/ disagreement or contradiction from text. For this task, normally supervised methods are used [1, 19]. Besides, the argumentation theory has been used to recognize the entailment and contradiction relationships between two texts in [11]. In [7], the quotations are classified to specific topics and polarity (pro/con) using language models in debate corpus. A probabilistic model is studied in [65] to extract different types of expressions including agreement/disagreement expressions. In our work in [84], we take a different approach by exploiting the special structure of CreateDebate. We also explore rich language units like N-grams

and dependency relations and illustrate their usefulness for viewpoint discovery. Part of the method uses sentiment analysis to extract opinions from text. This is built on top of a large body of existing work on opinion extraction, e.g. [12] and [103]. As the sentiment analysis component is not our main contribution, we do not review existing work along this direction in detail here. Interested readers can refer to [71].

Another closely related task is subgroup detection, i.e. to cluster users holding similar viewpoints (sides). There is a range of work that studies clustering-based approaches for the task [3, 14, 35]. [35] proposed to predict the polarity of interactions between users based on their textual exchanges. They defined a set of interaction features using sentiment analysis and applied supervised learning for polarity prediction. Both textual content and social interactions are studied in [54] to find opposing network from online forums.

2.1.3 User opinions

Not only user opinions toward other recipient is important to the task of mining user viewpoints, user opinions toward other opinion targets should also be taken into consideration. The work in [2] proposed to build discussant attitude profiles (DAP) from online posts, where a DAP is a vector that contains the attitudes of a discussant towards other discussants and a set of opinion targets. Based on the extracted DAPs, we can then cluster users into subgroups. In Chapter 3, we also extract opinions of users towards other users and opinion targets from posts, which are similar to DAPs. User opinions may be sparse, i.e., not all the users will pose their opinions on other users or opinion targets. To alleviate this problem, we further apply probabilistic matrix factorization to derive a low-rank representation from the raw opinion scores. Our comparison with DAP-based clustering shows that probabilistic matrix factorization can improve subgroup detection.

2.2 Micro-level stance prediction

Existing approaches for micro-level stance prediction have focused on taking advantage of the availability of different sources of information.

2.2.1 User arguments

There is a line of research work in micro-level stance prediction by employing linguistic features in user arguments. Specifically, Lin et al. [52] observe that people from opposing perspectives seem to use words in differing frequencies. Kim and Hovy [62] use unigrams, bigrams and trigrams for election prediction from forum posts. The work in [93] focuses on identifying stances (sides) in online debates by extracting useful linguistic features and making use of curated sentiment and argument lexicons. This work is complementary to that by Greene and Resnik [32], which focuses on syntactic packaging for stance prediction. The task here is essentially a supervised stance prediction task based on linguistic features where rich user generated texts are required.

2.2.2 User past stances

If we have a user's past stances, this problem is similar to item recommendation, where using a user's purchase history, his preference for a new item is to be predicted. In Chapter 5, we use a recommendation-based approach to predict a user's stance on an issue based on her past stances on related issues. This is based on the idea of collaborative filtering. Collaborative filtering [28] is a technique commonly used to alleviate the data sparsity problem in item recommendation. When applied to our stance prediction problem, the idea is to draw on clues from other issues on which a user's stance has been explicitly expressed as well as other users with similar patterns of stances. Collaborative filtering is a large research area, whose techniques can be readily applied to our user stance-issue matrix (see [94] for a survey). In our problem, we specifically consider PMF methods (proposed in [87]),

because of its successful use in real-world problems [9, 45, 55, 99].

The above method doesn't rely on any user arguments, but if we also have user arguments along with user stances, we can model such information into PMF framework. Specifically, we can use a latent Dirchlet allocation topic model [10] to reduce the dimensionality of the text, and combine text data with latent factors from the user stance matrix, grounding each dimension of the hidden factor using inferred topics. To achieve this, one may consider to adapt the models used in these studies [57, 99].

2.2.3 User social networks

Social behavior between users is another important factor for macro-level stance prediction. There are two types of social networks that are useful. One type is the topic-independent one which is usually more long-term and stable, e.g., networks of friends and enemies. The other type is topic-specific social network reflecting user relationship on a particular topic.

- Topic-independent social behaviors. Topic-independent social behavior is an important component used in several studies for prediction, recommendation and community detection tasks. For instance, the work in [55] proposed SocRec to extend a collaborative filtering framework with social network information to perform social recommendation. The basic idea of social recommendation is that a person's social network will affect personal behaviors like purchasing behavior on the Web. Similarly, there are studies on using trust-based recommendations, which incorporate trust network information [40, 41, 105].

- Topic-specific social behaviors. Topic-specific social network can be obtained from user interactions, e.g., like/dislike or thubsup/thumbsdown on other users' feedback for a specific topic, or agreement/disagreement inferred from texts exchanged between users. An important observation is that in forums or debate sites, users tend to dispute or agree with others on the debate

issues by replying directly to the commenter. The work [107] observed that users not only interact with others who share same views, but also actively engage with whom they disagree. It's thus important to incorporate such information for the task. In [31], an integrated model is proposed to jointly model user stances, user social network, and topic-specific social network. The model provides a hidden factor representation for each user, which can be used to cluster users or infer user stances. Similarly, in our work [82], we also incorporate user interactions to help the task of micro-level and macro-level stance prediction.

2.2.4 User attributes

Users with different "ideological" beliefs tend to take different stances or positions towards critical policies and sociopolitical issues. To collect user attributes that are close to "ideological" beliefs can help to reveal user stances towards different issues. One important type of attribute is user's political affiliation or political leaning. As shown in [30], a user who is associated with the Democratic Party in the U.S. tends to support *abortion* and oppose *gun rights*. In our work [82], we study the usefulness of different types of attributes from user's demographic information: party (e.g., republican, democrat), religion (e.g., catholic, christian), gender (e.g., male, female), status (e.g., single, married), education (e.g., in college, post grad.), and country (e.g., U.S., Singapore). Experiment shows that only these two types of attributes improve our base model: party and religion. This shows that those attributes related to "ideology" are useful for the task of stance prediction.

2.3 Macro-level stance prediction

There is a growing amount of research work related to analyzing publicly available social media data to infer user opinions in the larger population [13, 68, 72, 77, 95]. The task of macro-level stance prediction in online discussions is closely related

to these studies. But our notion of macro-level stance prediction is different from existing work in that we study online discussion forums which have well structured user arguments, opinions, and interactions on diverse topics, to seek to predict user stances on a wider variety of social topics. To the best of our knowledge, there is no previous work related to macro-level stance prediction in online discussions.

Note that we can resolve the task of macro-level stance prediction to micro-level stance prediction providing the following two types of information.

- User arguments. If all the users or the majority of the users have posted arguments on all the issues, we can simply predict any user's stance on an issue by examining all his arguments under the issue. We can consider work such as [52, 62, 93, 32] for this task.

- User past stances. In real-world scenario, we have data sparsity problem. For any specific issues, we only observe a few users participating and expressing their opinions. To alleviate this problem, we need to consider collaborative filtering, where the idea is to predict a user's stance based on other issues on which the user's stance has been explicitly expressed as well as other users with similar patterns of stances. We can also consider these work [31, 30, 57, 99] to model other information that is useful for the tasks, e.g., user social network and attributes. In our work [85], we propose an integrated model that jointly models user arguments, interactions, and attributes for the task of stance prediction. The model can be used for both micro-level and macro-level stance prediction.

2.4 User viewpoint summarization

For the task of user viewpoint summarization, we can borrow techniques from multi-document summarization which has been extensively studied in the NLP community, with most efforts on extractive summarization. Different features and

ranking strategies have been studied. Radev et al. [86] proposed to implement MEAD as a centroid-based summarizer by combining several predefined features like TF*IDF, cluster centroid and position to score sentences. Lin and Hovy [51] built the NeATS multi-document summarization system using term frequency, sentence position and stigma words. Nenkova et al. [67] proved that high-frequency words were significant in reflecting the focus of documents. Ouyang et al. studied the influence of different word positions in summarization [69]. Graph-based ranking algorithms nowadays are also successfully applied in summarization. This kind of algorithms takes global information into consideration rather than relying only on vertex-specific information, and therefore has been proved successful in document summarization. LexPageRank [17] is the representative work which is based on the PageRank [70] algorithm. Some methods have been proposed to extend the conventional graph-based models recently [59, 75, 98]. We have also studied using keyphrases [81] to summarize search results.

Integer Linear Programming (ILP) based framework is introduced as a global inference algorithm for multi-document summarization by [58], which considers informativeness and redundancy at sentence level. The framework is used by many studies for multi-document summarization [34, 39, 49, 88, 104]. In our study in Chapter 6, we consider this framework for user viewpoint summarization. [27] studies information and redundancy at a sub-sentence, "concept" level, modeling the value of a summary as a function of the concepts it covers. In their concept-based model, they use word bigrams weighted by the number of input documents in which they appear. We choose to build our solution based on ILP framework partially because in our preliminary analysis it outperforms other methods. Furthermore, it can be easily extended to incorporate more information. In our task, we make use of the information leant by our viewpoint discovery model in Chapter 3, and based on which we define a post relevance score by considering topic coverage and viewpoint distribution. We hypothesize that a good viewpoint specific summary should cover more viewpoint-specific topics and be relevant to the viewpoint.

Chapter 3

Modeling User Posting Behaviors

In this chapter, we study the problem of *viewpoint discovery* in forum threads. Since many controversial issues contain two contrastive viewpoints, e.g., for the issue "Do you believe God?", people will take either yes or no viewpoint, in this chapter, we focus on issues with two contrastive viewpoints. Nevertheless, the framework we developed can be potentially used to discover more than two viewpoints.

We focus on how to model each user's posting behaviors in terms of how he chooses topics and interacts with others when he expresses his viewpoints in online discussions. We will discuss two work related to this task.

We first introduce a latent variable model that jointly models user viewpoints, topics, and interactions for viewpoint discovery in online discussions. Our model takes three important characteristics into consideration, namely, user consistency, topic preference, and user interactions. The first one refers to the observation that a user's opinion on an issue usually remains unchanged during a certain time period. The second one models that users with different viewpoints tend to focus on different topics. This is close to a phenomenon called "framing" [96]. The last one refers to the observation that in a forum thread, like in conversations, users interact

with each other by commenting on each other's posts. Thus, modeling the agreement/disagreement expressions among users can help find different viewpoints.

As shown in above work, to tackle the task, it is important to exploit user posts that implicitly contain support and dispute (interaction) information. The challenge we face is how to mine such interaction information from the content of posts and how to use them to help identify stances. We proposes a two-stage solution based on latent variable models: an interaction feature identification stage to mine interaction features from structured debate posts with known sides and reply intentions; and a clustering stage to incorporate interaction features and model the interplay between interactions and sides for debate side clustering. Empirical evaluation shows that the learned interaction features provide good insights into user interactions and that with these features our debate side model shows significant improvement over other baseline methods.

3.1 Modeling Viewpoints, Topics and Interactions

Recently there has been some work on finding contrastive viewpoints from text. The model proposed by [74] assumes viewpoints and topics are orthogonal dimensions. Another model proposed by [18] assumes that documents are already grouped by viewpoints and it focus on identifying contrastive viewpoint words under the same topic. However, these existing studies are not based on interdependent documents like threaded forum posts. As a result, at least two important characteristics of threaded forum data are not considered in these models. (1) **User identity**: The user or publisher of each forum post is known, and a user may publish several posts in the same thread. Observed from our data sets, the same user's opinion on an issue usually remains unchanged. Hence posts published by the same user are likely to contain the same viewpoint. (2) **User interactions**. A thread is like a conversation, where users not only directly comment on the issue under discussion but also comment on each other's posts. Users having different viewpoints may

express their disagreement or even attack each other while users having the same viewpoint often support each other. The interaction expressions in forum posts may help us infer the relation between two users and subsequently infer the viewpoints of the corresponding posts.

In this chapter, we propose a novel latent variable model for viewpoint discovery from threaded forum posts. Our model is based on the following observations: First, posts with different viewpoints tend to focus on different topics. To illustrate this point, we first apply the Latent Dirichlet Allocation (LDA) model [10] on a thread about "will you vote Obama" and obtain a set of topics. This thread comes from a data set that has each user's viewpoint annotated. Using the ground truth viewpoint labels, we group all posts published by users with viewpoint 1 (or viewpoint 2) and compute the topic proportions. The two topic distributions are shown in Figure 3.1. We can see that indeed the two viewpoints each have some dominating topics. Our second observation is that the same user tends to hold the same viewpoint. In our model, we use a user-level viewpoint distribution to capture this observation, and the experiments show that it works better than assuming a global viewpoint distribution. Third, we define *positive interaction* as a user's reply to another user with agreement information, while *negative interaction* as a user replying to another user with disagreement. We observe that users with the same viewpoint are likely to have positive interactions while users with different viewpoints tend to have negative interactions. Using a sentiment lexicon, we can first predict the polarity of interaction expressions. We then propose a novel way to incorporate this information into the latent variable model. In summary, we capture the three observations above in a principled generative latent variable model. We present the details of our model in Section 3.1.1.

We use two tasks to evaluate our model. In the first task, we evaluate how well posts with different viewpoints are separated. In the second task, we evaluate how well our model is able to group users with different viewpoints. For both tasks, we compare our model with an existing model as well as a few degenerate versions of

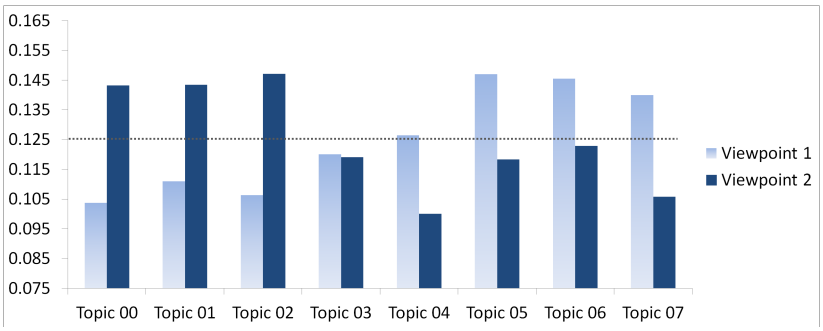

Figure 3.1: Topic distributions of two viewpoints for the thread "will you vote Obama?" The dotted line is the average topic probability.

our model. The results show that our model can clearly outperform the baselines in terms of three evaluation metrics. The experiments are presented in Section 3.1.3.

The contributions of our work are threefold: (1) We identify the importance of using user interactions to help infer viewpoints in forum posts. (2) We propose a principled latent variable model to jointly model topics, viewpoints and user interactions. (3) We empirically verify the validity of the three assumptions in our model using real data sets.

3.1.1 Model

Motivation

Before we formally present our latent variable model for viewpoint discovery, let us first look at the assumptions we would like to capture in the model.

Viewpoint-based topic distribution: The first assumption we have is that different viewpoints tend to touch upon different topics. This is because to support a viewpoint, users need to provide evidence and arguments, and for different viewpoints the arguments are likely different. To capture this assumption, in our model, we let each viewpoint have its own distribution of topics. Given the viewpoint of a post, the hidden topic of each word in the post is chosen according to the corresponding topic distribution associated with that viewpoint.

User identify: The second assumption we have is that the same user tends to talk from the same viewpoint, although there are also users who do not clearly have a

viewpoint. In our model, we assume that there is a user-level viewpoint distribution. For each post by a user, its viewpoint is drawn from the corresponding viewpoint distribution.

User interaction: An important difference between threaded forum posts and regular document collections such as news articles is that posts in the same thread form a tree structure via the "reply-to" relations. Many reply posts start with an expression that comments on a previous post or directly addresses another user. These interaction expressions may carry positive or negative sentiment, indicating an agreement or a disagreement. For example, Table 3.1 shows the interaction expressions from a few sample posts with words such as "correct," "agree," and "delusional," implying the polarity of the interaction expressions. The polarity of these interaction expressions can help us infer whether two posts or two users hold the same viewpoint or not. In our model, we assume that the polarity of each interaction expression can be detected. Details of how we perform this detection are in Section 3.1.1.

	Post
+	You are **correct**. Obama got into office w/ everything $\cdots$
	I **agree** with your post Dan. Obama is so $\cdots$
−	Most of your post is **delusional**, especially the part $\cdots$
	Are you **freaking** nutz? Palin is a BIMBO!

Table 3.1: Sample posts with positive (+) and negative(−) interactions.

While the way to capture the first two assumptions discussed above is fairly standard, modeling user interactions is something new. In our model, we assume that the polarity of an interaction expression is generated based on the viewpoint of the current post and the viewpoint of post(s) that the current post replies to. The intuition is that if the viewpoints are the same, we are more likely to see a positive interaction whereas if the viewpoints are different we are more likely to see a negative interaction.

Model description

We use the following notation to represent our data. We consider a set of forum posts published by U different users on the same event or issue, where user u ($1 \leq u \leq U$) has published N_u posts. Let $w_{u,n,l}$ ($1 \leq l \leq L_{u,n}$) denote the l-th word in the n-th post by user u, where $L_{u,n}$ is the number of words in the n-th post by user u. $w_{u,n,l}$ is represented by an index between 1 and V where V is the vocabulary size. Furthermore, we assume that some of the posts have user interaction expressions, where the polarity of the expression is known. Without loss of generality, let $s_{u,n} \in \{0, 1\}$ denote the polarity of the interaction expression of the n-th post by user u. In addition, for each post that has an interaction expression, we assume we also know the previous post(s) it replies to. (In the case when the current post replies to a user, we assume all that user's existing posts are being replied to.) We refer to these posts as the *parent posts* of the current post.

We assume that there are T topics where each topic is essentially a word distribution, denoted as ψ^t. We also assume that there are Y different viewpoints expressed in the collection of posts. For most controversial issues, Y can be set to 2. Each viewpoint y has a topic distribution θ^y over the T topics. While these T topics are meant to capture the topical differences between viewpoints, since these viewpoints are all about the same issue, there are also some words commonly used by different viewpoints. We therefore introduce a background topic ψ^B to capture these words. Finally, each user u has a distribution over the Y viewpoints, denoted as φ^u.

Figure 3.2 shows the plate notation of the complete model. We assume the following generation process in our model. When user u generates her n-th post, she first samples a viewpoint from φ^u. Let this viewpoint be represented by a hidden variable $y_{u,n}$. For the l-th word in this post, she first samples an indicator variable $x_{u,n,l}$ from a Bernoulli distribution parameterized by π. If $x_{u,n,l} = 0$, then she draws $w_{u,n,l}$ from ψ^B. Otherwise, she first samples a topic, denoted as $z_{u,n,l}$, according to $\theta^{y_{u,n}}$, and then draws $w_{u,n,l}$ from $\psi^{z_{u,n,l}}$.

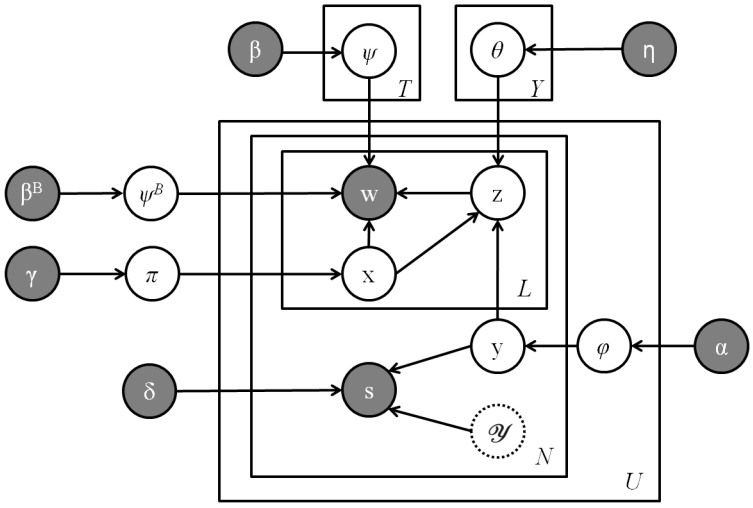

Figure 3.2: Plate notation of the Joint Viewpoint-Topic Model with User Interaction (JVTM-UI). The dotted circle for $\mathcal{Y}$ means the variables represented by $\mathcal{Y}$ are not new variables but a subset of the y variables.

Furthermore, if this post is a reply to a previous post or another user, she may first comment on the parent post(s). The polarity of the interaction expression in the post is dependent on the viewpoint $y_{u,n}$ and the viewpoints of the previous post(s). Let us use $\mathcal{Y}_{u,n}$ to denote the set of y variables associated with the parent posts of the current post. The user draws $s_{u,n}$ according to following distribution:

$$p(s_{u,n} = 1|y_{u,n}, \mathcal{Y}_{u,n}, \delta) = \frac{\sum_{y' \in \mathcal{Y}_{u,n}} \mathbb{I}(y_{u,n} == y') + \delta}{|\mathcal{Y}_{u,n}| + 2\delta},$$
$$p(s_{u,n} = 0|y_{u,n}, \mathcal{Y}_{u,n}, \delta) = 1 - p(s_{u,n} = 1|y_{u,n}, \mathcal{Y}_{u,n}, \delta), \tag{3.1}$$

where $\mathbb{I}(\cdot)$ is 1 if the statement inside is true and 0 otherwise, and $\delta > 0$ is a smoothing parameter.

Finally, we assume that ψ^B, ψ^t, φ^u, θ^y and π all have some uniform Dirichlet priors.

Inference

We use collapsed Gibbs sampling to estimate the model parameters. In the initialization stage of Gibbs sampling, for a reply post to a recipient, we initialize its corresponding reply polarity s according to all the labeled polarity of interaction

words. Specifically, if the majority of labeled interaction words are positive, we set $s = 1$, otherwise we set $s = 0$.

Let $\mathbf{Y}$ denote the set of all y variables, and $\mathbf{Y}_{\neg(u,n)}$ denote $\mathbf{Y}$ excluding $y_{u,n}$. Similar notation is used for the other variables. We sample $y_{u,n}$ using the following formula.

$$\begin{aligned}
& p(y_{u,n} = k|\mathbf{Y}_{\neg(u,n)}, \mathbf{Z}, \mathbf{S}, \mathbf{X}, \alpha, \eta, \delta) \\
\propto\ & \frac{p(y_{u,n} = k, \mathbf{Y}_{\neg(u,n)}|\alpha)}{p(\mathbf{Y}_{\neg(u,n)}|\alpha)} \cdot \frac{p(\mathbf{Z}|y_{u,n} = k, \mathbf{Y}_{\neg(u,n)}, \mathbf{X}, \eta)}{p(\mathbf{Z}_{\neg(u,n)}|\mathbf{Y}_{\neg(u,n)}, \mathbf{X}_{\neg(u,n)}, \eta)} \\
& \cdot p(\mathbf{S}|y_{u,n} = k, \mathbf{Y}_{\neg(u,n)}, \delta) \\
=\ & \frac{C^k_{u,\neg n} + \alpha}{C^{(\cdot)}_{u,\neg n} + Y\alpha} \cdot \frac{\prod_{t=1}^{T} \prod_{a=0}^{C^t_{u,n}-1} (C^t_{k,\neg(u,n)} + \eta + a)}{\prod_{b=0}^{C^{(\cdot)}_{u,n}-1} (C^{(\cdot)}_{k,\neg(u,n)} + T\eta + b)} \\
& \cdot p(\mathbf{S}|y_{u,n} = k, \mathbf{Y}_{\neg(u,n)}, \delta). \qquad (3.2)
\end{aligned}$$

Here all Cs are counters. $C^k_{u,\neg n}$ is the number of times we observe the viewpoint k from u's posts, excluding the n-th post, based on $\mathbf{Y}_{\neg(u,n)}$. $C^t_{u,n}$ is the number of times we observe topic t from user u's n-th post, based on $\mathbf{Z}_{u,n}$. And $C^t_{k,\neg(u,n)}$ is the number of times we observe topic t associated with viewpoint k, excluding user u's n-th post. Note that we need $\mathbf{X}$ to know which words are assigned to the background topic so we can exclude them for $C^t_{u,n}$ and $C^t_{k,\neg(u,n)}$. $C^{(\cdot)}_{u,\neg n}$ is the number of times we observe any viewpoint from u's posts, excluding the n-th post. $C^{(\cdot)}_{u,n}$ and $C^{(\cdot)}_{k,\neg(u,n)}$ are defined similarly.

The last term is further expanded as follows:

$$\begin{aligned}
& p(\mathbf{S}|y_{u,n} = k, \mathbf{Y}_{\neg(u,n)}, \delta) = p(s_{u,n}|y_{u,n} = k, \mathcal{Y}_{u,n}, \delta) \\
& \cdot p(\mathbf{S}_{\neg(u,n)}|y_{u,n} = k, \mathbf{Y}_{\neg(u,n)}, \delta). \qquad (3.3)
\end{aligned}$$

Here $p(s_{u,n}|y_{u,n} = k, \mathcal{Y}_{u,n}, \delta)$ is computed according to Eqn. (3.1). For the latter term, we need to consider posts which reply to user u's n-th post because the value

of $y_{u,n}$ affects these posts.

$$
\begin{aligned}
& p(\mathbf{S}_{\neg(u,n)}|y_{u,n}=k,\mathbf{Y}_{\neg(u,n)},\delta) \\
\propto & \prod_{(u',n'):y_{u,n}\in\mathcal{Y}_{u',n'}} p(s_{u',n'}|y_{u',n'},\mathcal{Y}_{u',n'},\delta).
\end{aligned} \tag{3.4}
$$

Next, we show how we jointly sample $x_{u,n,l}$ and $z_{u,n,l}$. We jointly sample them because when $x_{u,n,l} = 0$, $z_{u,n,l}$ does not need a value. We have the following formulas:

$$
\begin{aligned}
& p(x_{u,n,l}=1,z_{u,n,l}=t|\mathbf{X}_{\neg(u,n,l)},\mathbf{Z}_{\neg(u,n,l)},\mathbf{Y},\mathbf{W},\gamma,\eta,\beta,\beta^B) \\
\propto & \frac{C^1_{\neg(u,n,l)}+\gamma}{C^{(\cdot)}_{\neg(u,n,l)}+2\gamma}\cdot\frac{C^t_{y_{u,n,l},\neg(u,n,l)}+\eta}{C^{(\cdot)}_{y_{u,n,l},\neg(u,n,l)}+T\eta}\cdot\frac{C^{w_{u,n,l}}_{t,\neg(u,n,l)}+\beta}{C^{(\cdot)}_{t,\neg(u,n,l)}+V\beta},
\end{aligned} \tag{3.5}
$$

$$
\begin{aligned}
& p(x_{u,n,l}=0|\mathbf{X}_{\neg(u,n,l)},\mathbf{Z}_{\neg(u,n,l)},\mathbf{Y},\mathbf{W},\gamma,\eta,\beta,\beta^B) \\
\propto & \frac{C^0_{\neg(u,n,l)}+\gamma}{C^{(\cdot)}_{\neg(u,n,l)}+2\gamma}\cdot\frac{C^{w_{u,n,l}}_{B,\neg(u,n,l)}+\beta^B}{C^{(\cdot)}_{B,\neg(u,n,l)}+V\beta^B}.
\end{aligned} \tag{3.6}
$$

Here again the Cs are counters defined in similar ways as before. For example, $C^1_{\neg(u,n,l)}$ is the number of times we observe 1 assigned to an x variable, excluding $x_{u,n,l}$.

Interaction polarity prediction

The problem of detecting agreement and disagreement from forum posts is relatively new. One possible solution is to use supervised learning, which requires training data [1, 6, 19]. However, training data are also likely domain and language dependent, which makes them hard for re-use. For our task, we take a simpler approach and use a sentiment lexicon together with some heuristics to predict the polarity of interaction expressions. Specifically, we first identify interaction sentences following the strategies from [35]. We assume sentences containing mentions of the recipient of a post are interaction sentences. Next, we consider words within a text window of 8 words surrounding these mentions. We then use a subjectivity lexicon to label these words. To form an English lexicon, we combine three popular lex-

icons: the sentiment lexicon used by [38], Multi-Perspective Question Answering Subjectivity Lexicon by [102] and SentiWordNet by [8]. Since we also work with a Chinese data set, to form the Chinese sentiment lexicon, we use opinion words from HowNet[1] and NTUSD by [46]. To predict the polarity of an interaction expression, we simply check whether there are more positive sentiment words or more negative sentiment words in the expression, and label the interaction expression accordingly.

We would like to stress that since this interaction classification step is independent of the latent variable model, we can always apply a more accurate method, but this is not the focus of this work.

3.1.2 Models for Comparison

In our experiments, we compare our model, Joint Viewpoint-Topic Model with User Interaction (JVTM-UI), with the following baseline models.

JVTM: The model is shown in Figure 3.3(a), a variant of JVTM-UI that does not consider user interaction. Through comparison with it, we can evaluate the effect of modeling user interactions.

JVTM-G: We consider JVTM-G in Figure 3.3(b), a variant of JVTM which assumes a global viewpoint distribution. Comparison with it allows us to evaluate the usefulness of user identity in the task.

UIM: The third model we consider is a User Interaction Model (UIM) in Figure 3.3(c), where we rely on only the users' interactions to infer the viewpoints. We use it to evaluate how well viewpoints can be discovered from only user interaction expressions.

TAM: The last model we consider is the one by [74]. As input for TAM, it assumes each user has an article on an issue. In our data set, each user has a set of posts. We first concatenate all the posts by the same user into a pseudo document and then apply TAM.

[1] `http://www.keenage.com/html/e_index.html`

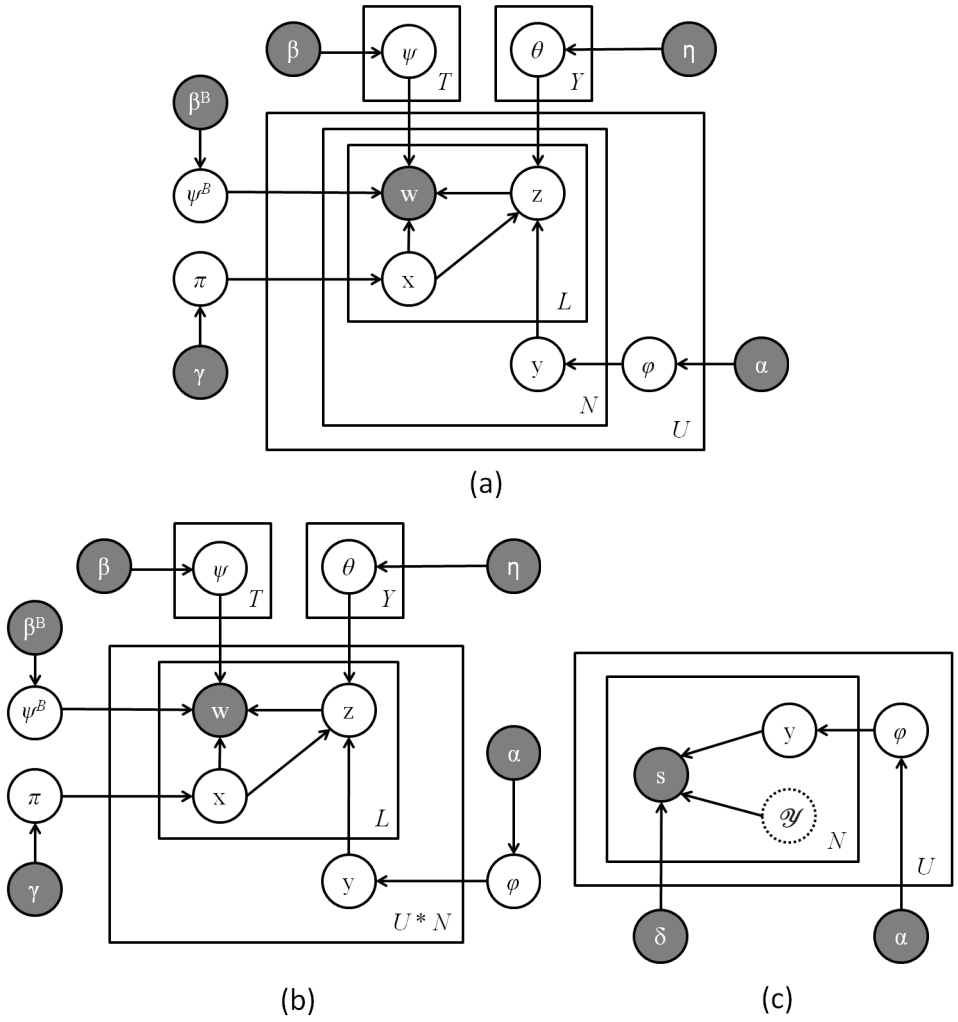

Figure 3.3: (a) JVTM: Joint Viewpoint-Topic Model. (b) JVTM-G: JVTM with a global viewpoint distribution. (c) UIM: User-Interaction Model.

3.1.3 Experiments and Analysis

In this section, we evaluate our model with a set of baseline models using two data sets.

Data Sets and Experimental Settings

We focus our work on finding users' viewpoints on a controversial issue, where we assume that there are two contradictory viewpoints. We use two data sets on controversial issues. The first data set comes from [2] and [35]. This data set originally was used for finding subgroups of users, so the annotations were done at user level, i.e. for each user there is a label indicating which subgroup he/she belongs to. We

Name	Issue	#Posts	#Users
EDS1	Vote for Obama	2599	197
EDS2	Arizona Immigration Law	738	59
EDS3	Tax Cuts	276	26
CDS1	Tencent and Qihoo dispute	30137	2507
CDS2	Fang Zhouzi questions Han Han	76934	1769
CDS3	Liu Xiang in London Olympics	29486	2774

Table 3.2: Some statistics of the data set.

use the top-3 mostly discussed threads with two subgroups for our study.

In reality, controversial issues are often discussed across threads. We thus constructed another large data set which contains more than one thread for each issue. We chose three hot issues from one of the most popular Chinese online forums — TianYa Club[2]. The three issues are "Fang Zhouzi questions Han Han"[3], "Tencent and Qihoo dispute"[4], and "Liu Xiang in London Olympics"[5]. All these issues triggered heated discussions on the forum and we found that most of the users were divided into two different groups.

We crawled the data set using the TianYa API[6]. The API allows users to issue queries and get threads most related to the queries. For each issue, we used entities involved in the event as queries and obtained 750 threads for each query. We then extracted all the posts in the threads. As there are users who posted irrelevant posts in the forum, we then filtered out those users who did not mention the entities or had fewer than 4 posts.

We refer to the first set of data in English as EDS1, EDS2 and EDS3, and the second set of data in Chinese as CDS1, CDS2 and CDS3. Some statistics of the resulting data set are shown in Table 3.2.

For all the models, we set $Y = 2$. We set $T = 10$ for the English data sets and $T = 40$ for the Chinese data sets. We run 400 iterations of Gibbs sampling as burn-in iterations and then take 100 samples with a gap of 5 to obtain our final

[2] http://en.wikipedia.org/wiki/Tianya_Club
[3] http://en.wikipedia.org/wiki/Fang_Zhouzi
[4] http://en.wikipedia.org/wiki/360_v._Tencent
[5] http://en.wikipedia.org/wiki/Liu_Xiang
[6] http://open.tianya.cn/index.php

results. We empirically set $\beta = 0.01$, $\beta^B = 0.1$, $\gamma = 10$ and $\delta = 0.1$ for our model on all the data sets. α and η are set through grid search where they take values in $\{0.01, 0.001\}$. For each data set, we choose the best setting for each model and report the corresponding results.

Identification of viewpoints

We first evaluate the models on the task of identifying viewpoints. For fair comparison, each model will output a viewpoint label for each post. For JVTM-UI, JVTM, JVTM-G and UIM, after we learn the model, each post will directly have a viewpoint assignment. For TAM we cannot directly get each post's viewpoint as the model assumes a document-level viewpoint distribution. To estimate each post's viewpoint in this model, we use viewpoint assignment at the word level learnt from the model. Then for each post, we label its viewpoint as the viewpoint that has the majority count in the post.

Ideally, we would like to manually label all the posts to obtain the ground truth for evaluation. Since there are too many posts, we only labeled a sample of them. For each issue, we randomly selected 150 posts to label their viewpoints. For each post, we asked two different annotators to label its viewpoint. We made sure that the annotators understand the issue and the two major viewpoints before they annotated the posts. Specifically, as the Chinese data sets are about some controversial issues around the entities involved, we then defined two major viewpoints as *support* and *not support* the entity who initiated the event. The entities of data set CDS1, CDS2 and CDS3 are *Fang Zhouzi*, *Tencent* and *Liu Xiang* respectively. For each given post, the annotators were asked to judge whether the post has expressed viewpoints and if so, what is its corresponding viewpoint. We measure the agreement score using Cohen's kappa coefficient. The lowest agreement score for an issue is 0.61 in the data set, showing good agreement. We then used the set of posts that were labeled with the same viewpoint by the two annotators as our evaluation data for all the models.

Since our task is essentially a clustering problem, we use *purity* and *entropy* to measure the performance [56]. Furthermore, we also use *accuracy* where we choose the better alignment of clusters with ground truth class labels and compute the percentage of posts that are "classified" correctly. For purity and accuracy, the higher the measure the better the performance. For entropy, the lower the measure is the better the performance.

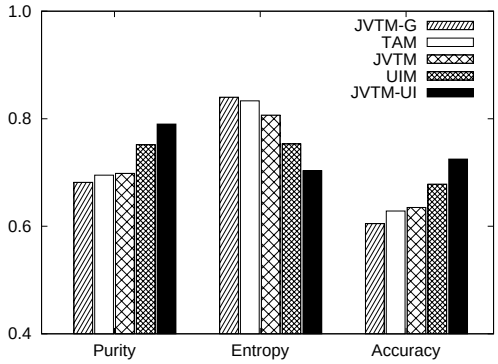

Figure 3.4: Averaged results of the models in identification of viewpoints.

We give an overview of all the averaged model results on the data sets in Figure 3.4. We observed that UIM performs relatively better than other methods except our model. This shows user interactions are important features to identify post viewpoints. Overall, our model has a better performance as it is with higher purity and accuracy, and lower entropy.

Table 3.3 shows the detailed results on the data sets. We perform the 2-tailed paired t-test as used by [2] on the results. All the result differences are at 10% significance level if not with further clarification. First, JVTM has a better performance over JVTM-G, which shows it is important to consider user identity in the task. Second, JVTM and TAM have similar performance on EDS1 and CDS2, but JVTM has a relatively better performance on EDS2, EDS3, CDS1 and CDS3. This shows it is helpful to consider each viewpoint's topic preference. Although as studied by [74], by only using unigram features, TAM may not be able to cluster viewpoints accurately. Our study shows that the results can be improved when adding each viewpoint's topic focus. Third, UIM has relatively better performance than

		JVTM-UI	UIM	JVTM	TAM	JVTM-G
EDS1	*P*	**0.77**	0.74	0.64	0.65	0.63
	E	**0.72**	0.76	0.90	0.92	0.94
	A	**0.77**	0.74	0.61	0.60	0.57
EDS2	*P*	**0.82**	0.78	0.68	0.65	0.64
	E	**0.69**	0.73	0.79	0.86	0.90
	A	**0.81**	0.78	0.68	0.68	0.65
EDS3	*P*	**0.79**	0.73	0.65	0.64	0.62
	E	**0.67**	0.79	0.88	0.89	0.87
	A	**0.79**	0.73	0.65	0.64	0.62
CDS1	*P*	**0.87**	0.83	0.83	0.82	0.82
	E	**0.61**	0.64	0.65	0.66	0.64
	A	**0.60**	0.58	0.59	0.58	0.57
CDS2	*P*	**0.71**	0.65	0.61	0.63	0.60
	E	**0.80**	0.85	0.92	0.95	0.96
	A	**0.71**	0.65	0.61	0.61	0.59
CDS3	*P*	0.78	0.78	0.78	0.78	0.78
	E	0.73	0.75	**0.70**	0.72	0.73
	A	**0.67**	0.59	**0.67**	0.66	0.63

Table 3.3: Results on viewpoint identification on the all data sets.

the other models, which demonstrates that user interactions alone can do a decent job in inferring viewpoints. Finally, our proposed model has the best performance across the board in terms of all three evaluation metrics. Note that, our proposed model significantly outperforms other methods at 5% significance level except at 10% significance level over JVTM model. This shows by jointly modeling topics, viewpoints and user interactions, our model can better identify posts with different viewpoints.

Identification of user groups

We also use another task to evaluate our model. The task here is finding each user's viewpoint and subsequently grouping users by their viewpoints. This task has been studied by [3], [14], [2] and [35]. For the English data set, the user-level group labels are provided by the original data set. For the Chinese data set, we randomly selected 150 users for each issue and manually labeled them according to their viewpoints as reflected by their posts. If a user's posts do not clearly suggest a viewpoint, we label her as neutral. Again we asked two human judges to do annotation. The agreement

scores are above 0.70 for all issues, showing substantial agreement. This score is higher than viewpoint identification, which suggests that it is easier to judge a user's viewpoint than a single post's viewpoint. We use the set of users who have got the same labels by the two human judges for our experiments. Similarly we compute *purity*, *entropy* and *accuracy* to evaluate the clustering results.

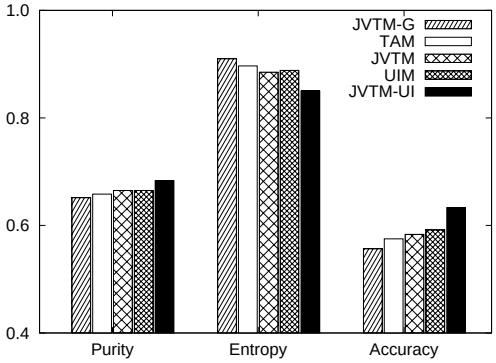

Figure 3.5: Averaged results of the models in identification of user groups.

Figure 3.5 shows the averaged results of all the models. Similar to previous experiment, our model has a better performance compared to the competing models.

		JVTM-UI	UIM	JVTM	TAM	JVTM-G
EDS1	*P*	0.67	0.67	0.67	0.67	0.67
	E	**0.85**	0.88	0.89	0.89	0.91
	A	**0.63**	0.59	0.58	0.59	0.57
EDS2	*P*	0.77	0.77	0.77	0.77	0.77
	E	**0.72**	0.76	0.74	0.75	0.76
	A	**0.62**	0.59	0.60	0.58	0.59
EDS3	*P*	**0.68**	0.63	0.61	0.61	0.58
	E	**0.90**	0.92	0.95	0.96	0.97
	A	**0.68**	0.63	0.61	0.58	0.57
CDS1	*P*	**0.64**	0.60	0.61	0.61	0.60
	E	**0.91**	0.97	0.96	0.96	0.97
	A	**0.61**	0.55	0.55	0.56	0.53
CDS2	*P*	0.69	0.69	0.69	0.69	0.69
	E	**0.83**	0.89	0.85	0.89	0.89
	A	**0.62**	0.57	0.56	0.58	0.54
CDS3	*P*	**0.67**	0.63	0.64	0.60	0.60
	E	**0.89**	0.91	0.92	0.93	0.96
	A	**0.64**	0.62	0.60	0.56	0.54

Table 3.4: Results on identification of user groups on all the data sets.

The results on the each data set are shown in Table 3.4. The tables show that

similar trends can be observed for the task of user group identification. We also perform the 2-tailed paired t-test on the results. We find our model significantly outperforms other models in terms of accuracy at 5% significance level, and purity and entropy at 10% significance level. Overall speaking, our joint model performed the best among all the models for this task for all three metrics. This shows that it is important to consider the topical preference of individual viewpoint, user's identify as well as the interactions between users.

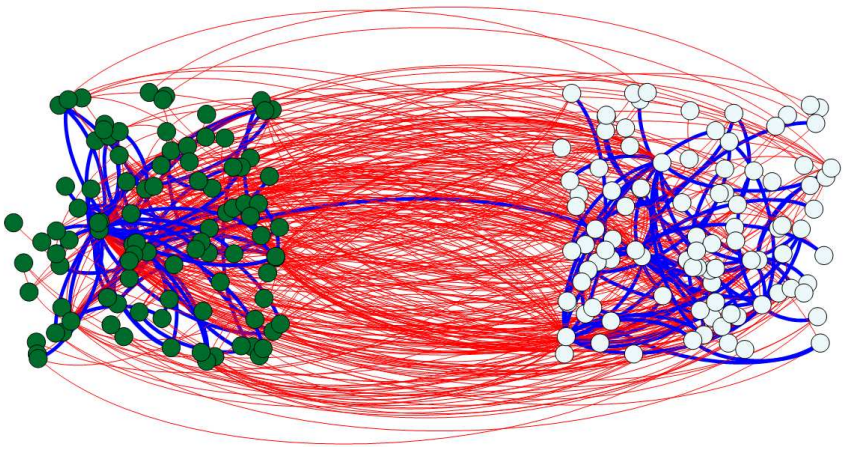

Figure 3.6: The user interaction network in a discussion thread about "will you vote obama." Green (left) and white (right) nodes represent users with two different viewpoints. Red (thin) and blue(thick) edges represent negative and positive interactions.

User interaction network

To gain some direct insight into our results, we show the user interaction network from one thread in Figure 3.6. Here each node denotes a user, and its color denotes the predicted viewpoint of that user. A link between a pair of users means these users have interactions and the interaction types have a dominant polarity. The polarities of these links are predicted using the interaction expressions and a sentiment lexicon, whereas the viewpoints of different users are learned by JVTM-UI, making use of the interaction polarities. The figure shows that clearly there are mostly positive interactions between users with the same viewpoint and mostly negative interactions between users with different viewpoints. Note that, our method to iden-

tify user interaction polarity is rule-based. As this step serves as a preprocessing step for our latent variable model, we can always use a more accurate method to improve the performances.

3.1.4 Summary

In this piece of work, we proposed a novel latent variable model for viewpoint discovery from threaded forum posts. Our model is based on the three important factors: viewpoint specific topic preference, user identity and user interactions. Empirical evaluation on the real forum data sets showed that our model could cluster both posts and users with different viewpoints more accurately than the baseline models.

3.2 Modeling Interactions Features

Online discussion forums are popular social media platforms for users to express their opinions and discuss controversial issues with each other. Most online discussion forums do not require users to explicitly indicate their stances or sides when they publish posts. Automatically clustering posts or users by their sides on an issue, also known as finding stances or sides, is an important task to help mine online opinions. In this chapter we focus on the task of clustering users/posts by sides on controversial issues.

So far, most existing work on finding viewpoints focuses on the topic differences in terms of the usage of words between documents with different viewpoints [18, 73]. Besides side-specific words and expressions, another important piece of information that is not yet well studied is user interactions, i.e. the interaction expressions exchanged between users. These interactions indicate if the users or posts support each other or disagree with each other.

This is especially evident when we look at online discussions, where user interactions are observed to be rich especially for those controversial discussion topics.

Examples include debate forums on social, political and cultural issues such as CreateDebate[7], where we find that the majority (∼80%) of the posts are interaction posts, i.e. posts that reply to other posts or users. Among these interaction posts, language units indicating user interactions are common.

Table 3.5 shows some sample posts from a debate page in CreateDebate. We observe that reply posts often contain interaction units that express opinions towards other users, e.g. unigrams like `right`, `wrong` and `foolishness`, trigrams like `how can we` and `how can you`. Another interesting finding is that many of these interaction related language units have polarities, and the polarity often indicates whether the sides of the two posts are the same. For example, positive unigrams like `yes` and `right` are used between User A and User C, who are on the same side, whereas negative unigrams like `wrong` and `foolishness` are used between User A and User B, who are on different sides. This is also true for trigrams. For example, `how can you` tends to be used between users with different sides like User K and User L. This also shows that to model interaction polarity, one may need to consider N-grams too. Besides this, one may find dependency relations can also be used to infer interaction polarity. For example, in the sentence `you cannot even prove it`, a dependency relation like ¬`nsubj(prove,you)`[8] indicates a negative interaction while by solely looking at N-grams, it is not clear to infer its polarity. In summary, these sample posts suggest that it is important to use interaction-related language units to infer interaction polarity and model the interplay between interactions and sides for side clustering or prediction. For the rest of the chapter, we use *interaction features* to refer to these interaction-related language units including N-grams and dependency relation tuples.

There have been some recent advances in analyzing user interactions, e.g. to extract agreement and disagreement expressions [65, 64] and to infer user relations by looking at their textual exchanges [35]. These approaches require either sentiment

[7] `http://www.createdebate.com/`

[8] `nsubj(prove,you)` means `you` is the subject of `prove` and ¬ means one of the words has been negated.

Debate: ***Does God Exist?***	
"Yes" Side (Side 0)	"No" Side (Side 1)
User A: Theists: I believe God exists. Atheists: I believe God doesn't exist. Both rely on belief . . . (*Side 0*) ↻User B (*Disputed*): Whoops. **wrong**. more like "I don't believe in god." . . . it is **gullibility** and **foolishness**. . . (*Side 1*) ↻User A (*Disputed*): . . . You BELIEVE there's no God. you cannot even prove it (*Side 0*) ↻User C (*Disputed*): **Yes**, that is **right**. Believe or not believe that is depend on the thinking and belief of everybody. I don't care anymore . . . (*Side 1*)	User J: If there is no evidence leading up to a God, I dont believe. . . (*Side 1*) ↻User K (*Disputed*): . . . if God is the very fabric of the universe and existence itself, **how can we** prove that it doesn't exist??? have no choice but to accept it (*Side 0*) ↻User L (*Disputed*): So **how can you** argue for something that you cannot even interact with on a comparable level? (*Side 1*) ↻User M (*Supported*): Question:Why did the crusades happen? Answer: god told the people to kill muslims . . . (*Side 1*)

Table 3.5: Sample posts on the debate "Does God Exist?"

lexicons, which may not be designed for user interactions, or labeled training data, which is labor-intensive to create. In the interaction feature identification stage, we propose a different approach to analyze user interactions. We observe that in some online forums such as CreateDebate, the intention of a reply post, i.e. whether it is supporting or disagreeing with the previous post, is clearly indicated. The side of each post is also known. When we have such rich structural information about the debate posts, we can make use of these labels to infer interaction features. In particular, we propose an *Interaction Model* (IM) to mine interaction features from these labeled debate posts. Another advantage of our model is that we adopt rich language features instead of the traditional "bag-of-words" features, which helps us gain more insights into user interactions.

After we mine the interaction features from the labeled debates, in the clustering stage, we propose a *Debate Side Model* (DSM) for side clustering by incorporating the learned interaction features. DSM can be applied for any forum threads whose reply structure is evident but side labels and interaction polarities are unknown. DSM segregates the interaction features from side-specific features to aid our side clustering tasks. It also automatically infers the interaction polarities of reply posts

and considers the interplay between interactions and sides. As demonstrated in our experiments, our two-stage solution yields better performance than all other competing methods we consider for evaluation.

Our contributions are: (1) To analyze user interactions, while most existing approaches require either sentiment lexicons or labeled training data, we propose to mine interaction features from structured debate posts with known sides and reply intentions. Experiment results show our extracted interaction features are insightful. (2) We propose a new debate side model to cluster posts or users by sides for general threaded discussions. The model incorporates two important factors: interaction features and the interplay between interactions and sides. (3) Empirical evaluation shows the advantages of our proposed models and the benefits of considering the aforementioned two factors.

3.2.1 Stage One - Model Interactions

In this section, we discuss our first stage to show how to model interaction features from CreateDebate data.

Data property. As presented in Table 3.5, a reply post in CreateDebate has three pieces of information: the debate side, the recipient post, and the reply intention – "support," "dispute" or "clarify." We treat "support" and "clarify" as a positive interaction (P) while "dispute" as a negative interaction (N).

We study different types of language features to represent posts.

Bag-of-Words. This simply considers all the unigram words.

N-grams. This considers all the N-grams inside a post, where $N \in \{2, 3\}$. For a sentence: `you cannot prove`, besides all the unigrams, we have three N-gram features: `you cannot`, `cannot prove` and `you cannot prove`.

Dependency Relations. As syntactic information can improve the accuracy of sentiment models [43], we thus consider adding syntactic features to our model. For each post, we use the Stanford parser [44] to get its dependency relations. For ex-

ample, for the above sentence, we will get these relations:`nsubj(prove,you)`, `aux(prove,can)` and `neg(prove,not)`[9]. This representation is referred to as *full-tuple* representation. As this representation has low generalization power, *split-tuple* representation is used in [32, 43]. In split-tuple representation, each dependency relation will be split into two relations. For example, `nsubj(prove,you)` will be split to `nsubj(prove,*)` and `nsubj(*,you)`.

Negation. We also consider negation features as studied in [74]. For a relation tuple `rel(a,b)`, if either a or b is negated, we rewrite the tuple as `¬rel(a,b)`; for the above sentence, we have `¬nsubj (prove,you)` and `¬aux(prove,can)` as features in full-tuple representation, and based on which we can re-build split-tuple features. With the three types of language features defined above, each post is now represented as a bag of these features. In the probabilistic model we present below, we use "*word*" to refer to any of these features, i.e. a word can be a unigram, an N-gram, or a negated or non-negated dependency relation.

Interaction Model

Our Interaction Model is a generative latent variable model that takes into consideration the data structure of the posts from CreateDebate to model interaction features. Specifically, we assume three types of words in debate posts.

Thread-specific word distribution ϕ^{T}. This models words specific to a debate thread. Taking the debate "Does God Exist?" for example, words such as `god` and `existence` can be thread-specific.

Side-specific word distribution ϕ^{S}. This models those words specific to each side of a debate. The intuition is that users from different sides tend to have different focuses and usage of words, which is close to a phenomenon called "framing" [54, 96]. For example, we find users on the "Yes" side talk more about the `bible` and use words like `religion` and `belief`. On the other hand, those on the "No" side

[9] `you cannot prove` will be tokenized as `you can not prove` by using the Stanford parser [44]

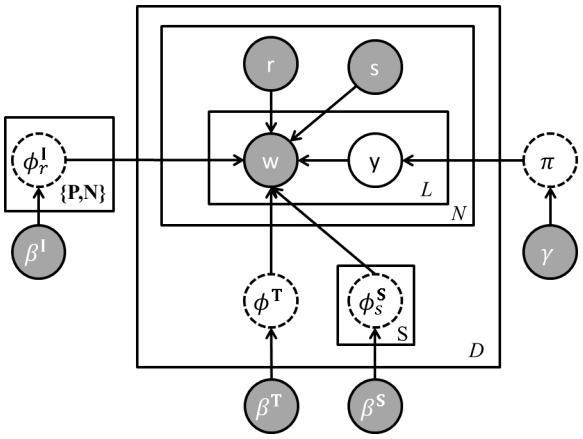

Figure 3.7: Interaction Model for modeling interaction words using the CreateDebate data. Dashed variables will be collapsed out in Gibbs Sampling.

tend to use words like `logic`, `rationality` and `science`.

Interaction word distribution ϕ^{I}. If a post is a reply to another post, it is highly possible that we observe some interaction words. For example, `yes`, `right` and `wrong` as shown in Table 3.5.

Assuming we have a set of debate threads where each thread focuses on a particular debate topic. Each thread has a set of posts where each post has a side. We use $s_{d,n} \in \{+, -\}$ to denote the side of the n-th post of the d-th thread, $r_{d,n} \in \{\mathsf{P}, \mathsf{N}\}$ to denote the relation of this post to its parent post[10]. We assume that the words in each post are generated from the three types of word distributions as described above, i.e. ϕ^{T}, ϕ^{S}, and ϕ^{I}. The plate notation of the model is in Figure 3.7 and the generative process is in Figure 3.8.

Beyond interaction words

The interaction words we are interested in are mostly opinion words. After some preliminary experiments, we find it more effective to only allow certain words to be assigned as interaction words. This treatment is similar to [18] where the authors assume opinion words are adjectives, verbs and adverbs.

In our study, we approximate this step by considering three types of features: (1) All the adjectives and adverbs. These adjectives and adverbs are identified by

[10]Both $s_{d,n}$ and $r_{d,n}$ are evident from CreateDebate structure.

- Draw selector distribution $\pi \sim \text{Dir}(\gamma)$
- For each interaction type $r \in \{\mathsf{P}, \mathsf{N}\}$
  - Draw $\phi_r^{\mathsf{I}} \sim \text{Dir}(\beta^{\mathsf{I}})$
- For the d-th thread ($d = 1, 2, \cdots, D$)
  - Draw $\phi_d^{\mathsf{T}} \sim \text{Dir}(\beta^{\mathsf{T}})$
  - Draw $\phi_{d,s}^{\mathsf{S}} \sim \text{Dir}(\beta^{\mathsf{S}})$ for each side s
  - For the n-th post ($n = 1, 2, \cdots, N_d$)
    - For the l-th word ($l = 1, \cdots, L_{d,n}$)
      - Let $s = s_{d,n}, r = r_{d,n}, y = y_{d,n,l}$, and $w = w_{d,n,l}$
      - Draw y from Multi(π)
      - Draw w as follows:

$$w \sim \begin{cases} \text{Multi}(\phi_d^{\mathsf{T}}) & \text{if } y = 0 \\ \text{Multi}(\phi_{d,s}^{\mathsf{S}}) & \text{if } y = 1 \\ \text{Multi}(\phi_r^{\mathsf{I}}) & \text{if } y = 2 \end{cases}$$

Figure 3.8: The generative process of the interaction model for CreateDebate. "Dir" and "Multi" stand for Dirichlet and Multinominal respectively.

the Stanford POS tagger. Note that these are unigrams; (2) Words that appear in one of the following opinion lexicons: the sentiment lexicon used in [38], Multi-Perspective Question Answering Subjectivity Lexicon [102] and SentiWordNet [8]; (3) Any N-grams containing at least one word from the above two types. We also consider N-grams that contain pronouns and verbs as these are oftentimes associated with opinions as studied in [65]; (4) Any negated and non-negated dependency relation tuples with at least M occurrences in the data set, e.g. `prep_with(agree,*)` and ¬`prep_with(agree,*)`. We empirically set M to 5.

We use collapsed Gibbs sampling to obtain samples of the hidden variable assignment and to estimate the model parameters from these samples. With Gibbs sampling, we can deduce the following estimation for interaction word distribution:

$$\phi_{r,w}^{\mathsf{I}} = \frac{C_{r,w}^{\mathsf{I}} + \beta^{\mathsf{I}}}{\sum_{w=1}^{V} C_{r,w}^{\mathsf{I}} + V\beta^{\mathsf{I}}}. \qquad \text{interaction-word distribution.} \tag{3.7}$$

where V is the vocabulary size, $C_{r,w}^{\mathsf{I}}$ is the number of times that word w co-occurs with interaction r. The interaction word distribution $\phi_{i,w}^{\mathsf{I}}$ is used in the later stage to infer interaction polarities of posts.

3.2.2 Stage Two - Cluster Sides

Clustering the posts or users participating in a debate based on their sides can help us understand the contentions and user groups exhibited in the debate. These two tasks are different as users may not always explicitly express their opinions in a post, nor do they always hold the same side throughout all the posts. The tasks are especially useful for understanding online debates with unknown side information for posts. We propose a generative model which can be applied for any forum settings for these tasks. Our model is adapted from the model in our previous work [80], specifically we borrow these two assumptions: user consistency and interplay between interactions and sides.

User Consistency: The same user tends to be on the same side for a given debate, although there are also users who do not have a clear side. In our model, we assume that there is a user-level side distribution. For each post by a user, its side is drawn from the corresponding side distribution.

Interplay between interactions and sides: An important difference between debate posts and regular document collections such as news articles is that posts in the same thread form a tree structure via the "reply-to" relations. The interaction polarity reflects the two users' side relation. Typically, if the sides are the same, we are more likely to see a positive interaction whereas if the sides are different we are more likely to see a negative interaction.

Note that we didn't consider user-topic preference as it shows little improvement over these two assumptions. And different from [80], the model will automatically infer the interaction polarity by using the learnt interaction words. Below we discuss the model in detail.

Debate Side Model

Our Debate Side Model is a generative model which assumes that interaction word distribution ϕ_r^{I} is known. Given the learned interaction word distributions, we also

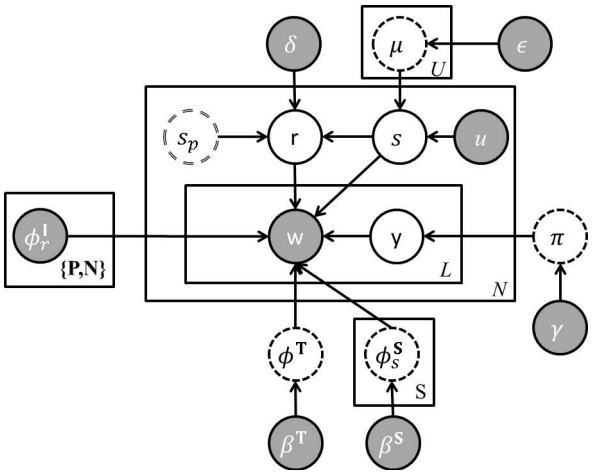

Figure 3.9: Plate notation for the Debate Side Model (DSM) on a given debate. Dashed variables will be collapsed out in Gibbs sampling. Double bordered dash variables are not new variables but a subset of the s variables.

assume a selector y which takes three values that correspond to thread-specific words, side-specific words and interaction words. For a given debate, we assume the polarities of the reply relations between posts and the side information of each post are unknown. We assume the same generative process to draw the words as in Figure 3.8. The plate notation of DSM is in Figure 3.9 and the generative process for the reply relations and the side information for the n-th post is shown in Figure 3.10.

- Draw $\mu_u \sim \epsilon$ for each participating user u
- For the n-th post
  - Let u_n be the author of the post
  - Draw side $s_n \sim \text{Multi}(\mu_{u_n})$
  - If the current post is a reply post, let s_n^{p} denote the parent post's side. Draw interaction type r_n from $p(r|s_n, s_n^{\mathsf{p}})$

Figure 3.10: The generative process of the debate side model.

The polarity of the interaction expression in the post is dependent on the side s_n of the post itself and the side s_n^{p} of the parent post. The user draws r_n according to the following distribution:

$$\begin{aligned} p(r_n = 1|s_n, s_n^{\mathsf{p}}, \boldsymbol{\delta}) &= \frac{\mathbb{I}(s_n^{\mathsf{p}} == s_n) + \delta_1}{1 + \delta_1 + \delta_0}, \\ p(r_n = 0|s_n, s_n^{\mathsf{p}}, \boldsymbol{\delta}) &= 1 - p(r_n = 1|s_n, s_n^{\mathsf{p}}, \boldsymbol{\delta}), \end{aligned} \tag{3.8}$$

where $\mathbb{I}(\cdot)$ is 1 if the statement inside is true and 0 otherwise, and δ_1,δ_0 are smoothing parameters. $r_n = 1$ when interaction is positive and 0 otherwise.

We also use Collapsed Gibbs sampling to estimate the parameters in our model. The main challenge in derivation is to consider the interplay between the side variable s and interaction type r, similar to the one studied in [80]. With Gibbs sampling, we can deduce the following estimation:

$$\phi_w^{\mathsf{T}} = \frac{C_w^{\mathsf{T}} + \beta^{\mathsf{T}}}{\sum_{w=1}^{V} C_w^{\mathsf{T}} + V\beta^{\mathsf{T}}}. \quad \text{thread-word distribution} \tag{3.9}$$

$$\phi_{s,w}^{\mathsf{S}} = \frac{C_{s,w}^{\mathsf{S}} + \beta^{\mathsf{S}}}{\sum_{w=1}^{V} C_{s,w}^{\mathsf{S}} + V\beta^{\mathsf{S}}}. \quad \text{side-word distribution} \tag{3.10}$$

Models for Comparison

We study both degenerate models and existing approaches for comparison.

DSM-1: The model is presented in Figure 3.11(a). By comparing it to DSM, we evaluate the importance of adding the interplay between interactions and sides.

DSM-2: The model is presented in Figure 3.11(b). Comparing it to DSM-1, we evaluate the importance of adding interaction words into the model.

DSM-SA: The model is the same with DSM except that the learned interaction words are replaced by opinion lexicons. By comparing it to DSM, we evaluate whether our learned interactions words can be replaced by simple opinion lexicons.

TAM: The Topic-Aspect Model (TAM) was proposed in [74, 73] for finding viewpoints without any learned interaction features. By comparing it with DSM-2, we can evaluate the necessity of adding interaction features.

K-Means: For each post or user, we use vector space model to build a vector on it using all the features. We then use K-Means to cluster them. By comparing it with DSM-2, we can see the effectiveness of considering side-specific features.

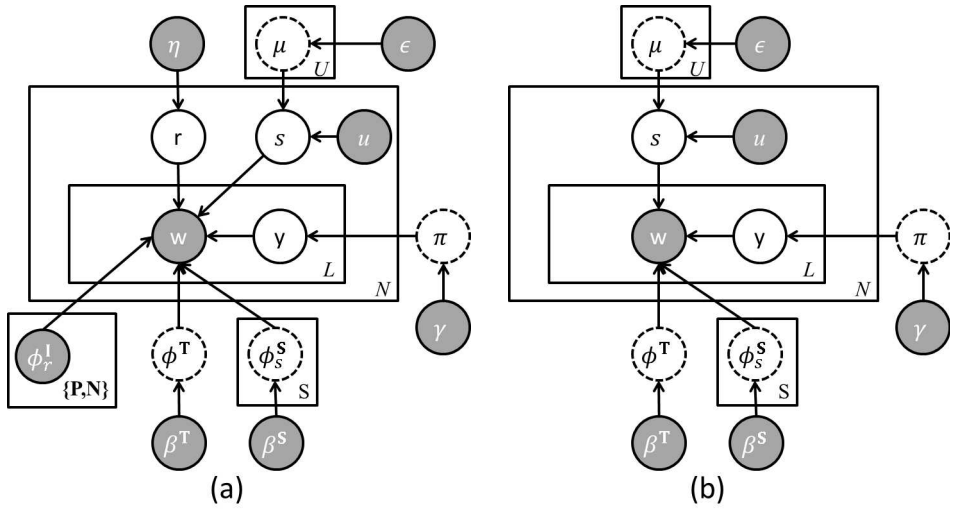

Figure 3.11: (a) DSM-1: A side clustering model that does not consider the interplay between interactions and sides. (b) DSM-2: A side clustering model that does not consider user interactions. Dashed variables will be collapsed out in Gibbs sampling.

3.2.3 Experiments

In this section, we first explain our corpus, then design our experiments to evaluate both Interaction Model and Debate Side Model.

Data

We crawled the top-80 popular debates from CreateDebate. We use top half of the debates for learning the interaction features using our Interaction Model and the other half for evaluating the Side Clustering Model. The statistics are shown in Table 5.2.

	A. Post#	A. User#	V_W	V_F	Inter.%
Train	273.6	66.2	32,677	40,874	0.81
Test	168.7	45.3	21,186	29,414	0.80

Table 3.6: Some statistics of the data set. A. Post# and A. User# refer to average number of posts and users for a thread, V_W and V_F are the total number of unique words and features. Inter.% stands for the percentage of reply posts.

For all the models, we set $S = 2$ for all debates. The model results are averaged from 10 runs, where for each run we perform 500 iterations of Gibbs sampling in

the burn-in stage and take 20 samples with a gap of 5 iterations to obtain our final results. We set δ_1 to 0.4 and δ_0 to 0.6 for our model[11]. For the other parameters ϵ, β^{T}, and β^{S}, we select the optimal setting based on average of 10 runs where they take values from $\{0.1, 0.01\}$. We use the same setting for our method and the baseline models (DSM-1, DSM-2 and DSM-SA). For TAM, we use the same setting in the paper [73]. We also vary the parameters in the above way and report the optimal results. For K-Means, we set $K = 2$ and use Euclidean distance.

Qualitative analysis of interaction features

We first qualitatively analyze the interaction features discovered by our Interaction Model.

P_W	N_W	P_NG	N_NG	P_DEP_NEG	N_DEP_NEG
good	choose	i agree	never like	prep_with(agree,*)	¬aux(*,do)
agree	easy	agree with	you have no	nn(lol,*)	¬aux(*,is)
affirm	knowledge	i do	you are not	advmod(agree,*)	amod(*,natural)
love	actually	i agree with	how is	dep(agree,*)	dobj(provide,*)
better	book	thank you	no longer	admod(*,well)	advmod(*,actually)
children	logical	not believe	are you	prep_to(*,religion)	¬xcomp(need,*)
winning	against	we can	you do	advmod(needed,*)	cop(irrelevant,*)
terrorism	irrelevant	believe in	what you	nsubj(*,love)	aux(arguing,*)
true	belief	even though they	you seem to	amod(*,good)	¬nsubj(is,*)
destroy	failed	do believe	is actually	advmod(feel,*)	¬dobj(have,*)

Table 3.7: Top unigrams(W), N-gram (NG), dependency relation and negation features for P(positive) and N(negative) interactions. As negation features are added directly into dependency relation features, we use DEP_NEG to denote their combinations.

We present top interaction features in Table 3.7. We find that: (1) The positive interaction words are often with positive sentiment like `true` and `love`, while the negative interaction words contain negative words like `against` and `irrelevant`. This shows the extracted interaction words are meaningful.[12] (2) N-grams tend to feature more identifiable expressions. E.g., `i agree` and `agree with you` show

[11] δ_1 and δ_0 represent to what extent we believe users from the same side tend to have positive interaction and from different sides with negative interaction. We set $\delta_1 + \delta_0 = 1$ and vary δ_1 from 0.3 to 0.7 with an interval of 0.1. We do not observe significantly result differences for our model. But we find $\delta_1 < 0.5$ yields relatively better results. This correlates to our data set property, as we observe users with different sides almost always "dispute" to each other, while users with the same side do not always "support" or "clarify" each other.

[12] The interaction words are not all sentiment words, e.g. `actually`. Although not shown in table, we observe many other none sentiment words, e.g. `spiritually` and `yep` for positive interactions and `simply` for negative interactions.

clear positive opinions, while `you have no` and `you are not` are oftentimes associated with negative opinions. (3) Positive dependency relations to be meaningful as well, e.g. `prep_with (agree,*)` and `nn(lol,*)` are popular for positive interactions. Moreover, we observe many negated expressions, e.g. `¬aux(*,do)` and `¬aux(*,is)`. In summary, with N-grams, dependency relation and negation features we can find more reasonable positive and negative interaction features to help infer interaction polarity.

Identification of interaction polarity

Now we use the task of predicting whether a reply post is positive (i.e. support or clarify) or negative (i.e. dispute) to evaluate the quality of the interaction features discovered by the Interaction Model. We use the Debate Side Model to learn the interaction polarity for each reply post by using the learned interaction features in the first stage. We then evaluate its accuracy. Recall that in CreateDebate, reply polarity is explicitly given in each reply post, which is used as the ground truth.

We consider a sentiment lexicon approach as baseline, where the approach will estimate the post's reply polarity by looking at the majority polarity of all the sentiment words with the post. If most of the sentiment words are positive, then it predicts the interaction as positive, otherwise predicts as negative.

We use *purity* (the higher the better) and *entropy* (the lower the better) to evaluate the performance of post clustering. We further use *accuracy* obtained by choosing the best alignment of clusters with the ground truth class labels and computing the percentage of users that are classified correctly.

As shown in Table 3.8, by using sentiment lexicon, an accuracy of 0.59 can be achieved. With bag-of-word representation, our model outperforms the lexicon based method. By sequentially adding N-grams, dependency relation and negation features, the model results can be further improved. Furthermore, we observe that there exists negative interactions among users with the same side, and these posts are often used to show partial disagreement with the recipients and do not contain

many negative interaction words. In this case, these negative interactions will be mis-labeled as positive interactions. This shows it will be interesting to further analyze the language usage in negative interactions to further improve accuracy; we leave it as future work.

Methods	Accuracy	Purity	Entropy	F_1-W
Lexicon	0.592	0.823	0.654	0.621
DSM + F_{W}	0.619	**0.836**	0.643	0.650
DSM + $F_{\text{W, NG}}$	0.622	**0.836**	0.643	0.662
DSM + $F_{\text{W, NG, DEP}}$	0.625	**0.836**	**0.641**	0.671
DSM + $F_{\text{W, NG, DEP, NEG}}$	**0.664**	**0.836**	**0.641**	**0.689**

Table 3.8: Interaction polarity identification. DSM + $F_{\text{W, NG, DEP, NEG}}$ stands for DSM with bag-of-words, N-gram, dependency relation and negation features. F_1-W is the average score of F_1 measure scores on positive and negative interaction prediction weighted by their proportions.

In summary, our method shows advantage over the lexicon based approach, and by adding N-grams, dependency relation and negation features its performance can be further boosted. We note that our focus here is not to propose a perfect solution to identify user interaction polarity, but rather to use a reasonable solution to identify interaction features to help the following side clustering tasks.

Clustering posts by sides

We evaluate our Debate Side Model on the task of post side clustering. In this task, for fair comparison, each model should output a side label for each post. For our model, the two degenerate models (DSM-1 and DSM-2) and DSM-SA, each post has a side label. For TAM, the side of a post is the one that has the majority word count in the post. For K-Means, we use the cluster index as the side of a post. We again use *purity*, *entropy* and *accuracy* to evaluate the performance of post clustering.

Results: We present the average results of all the debates in Table 3.9. We perform Wilcoxon signed-rank test on the performance of all debates. Our findings are the follows. (1) The fact that DSM-2 significantly outperforms K-Means at

5% significance level in terms of all the criteria shows it is importance to separate side-specific words apart from thread-specific words. (2) DSM-1 significantly outperforms DSM-2 at 10% significance level in terms of all the criteria. This shows by bringing in interaction features we can better identify sides. (3) We find modeling the interplay between interactions and sides in the DSM model can further boost the performances, as DSM significantly outperforms DSM-1 at 1% significance level. (4) DSM shows significantly better results than DSM-SA, at 5% significance level, which shows using standard opinion lexicons is not sufficient for the task. In summary, our DSM model shows significantly better performance than other baseline models, at least 5% significance level. This result clearly shows the effectiveness of considering interaction words and the importance of modeling the interplay between interactions and sides.

	DSM	DSM-1	DSM-2	DSM-SA	TAM	K-Means
A	**0.664**‡	0.636	0.619	0.637	0.548	0.563
P	**0.702**‡	0.675	0.666	0.678	0.557	0.566
E	**0.813**‡	0.860	0.869	0.851	0.982	0.973

Table 3.9: Post side clustering results. ‡ means the result is better than others in the same column at 5% significance level measured by Wilcoxon signed rank test. *A,P,E* denote Accuracy, Purity and Entropy respectively.

Clustering users by sides

We also use the task of finding each user's side and subsequently grouping users by their sides to evaluate our model. This task has been studied by [2, 3, 14, 35]. For fair comparison, each model should output a side label for each user. For our model and the two degenerate models, each user has a side distribution and we select the side which has the higher probability as the user's side. For TAM, we aggregate all the posts from a user to form a "document" and choose the side that has the majority word count in the "document" as this user's side. For K-Means, we use all posts of a user to form a feature vector and use the cluster index as the user's side. Similarly we use *purity*, *entropy* and *accuracy* to evaluate the clustering results.

	DSM	DSM-1	DSM-2	DSM-SA	TAM	K-Means
A	**0.622**‡	0.564	0.569	0.550	0.594	0.563
P	**0.618**†	0.591	0.592	0.577	0.609	0.566
E	**0.942**$^{\diamond}$	0.955	0.955	0.968	**0.942**$^{\diamond}$	0.973

Table 3.10: User side clustering results. ‡ means the result is better than others in the same column at 5% significance level measured by Wilcoxon signed rank test, † is at 10% level, $^{\diamond}$ means the results is better than others without this symbol in the same column at 5% significance level. *A,P,E* denote Accuracy, Purity and Entropy respectively.

Results: We present the average performance of all the debates in Table 3.10. We again perform Wilcoxon signed-rank test on the performance of all debates. Our findings are similar to the evaluation at the post level. As the number of users is much smaller than the number of posts, we find the result differences are not as significant as in post-level evaluation. Nevertheless, we still observe a better performance by DSM than other baseline models in terms of accuracy and entropy at 10% significance level. TAM shows a similar performance with DSM in terms of purity. By comparing DSM with DSM-1, we can still see the benefits of considering the interplay between interactions and sides. Again, we can still observe DSM significantly outperforms DSM-SA, at 5% significant level, which further shows the advantage of learned interaction features over standard opinion lexicons. All these results drive home that to consider interaction words and model the interplay between interactions and sides can help the debate side clustering task.

Impact of balanced data sets

In this section, we will present an analysis on the balance level of our data sets to further evaluate the robustness of these models. Not all debates are balanced on the two sides. To see whether our model has a robust performance on unbalanced data sets, we split our data sets based on different balance levels and compare the accuracy of all models for both post and user clustering tasks. The balance level of a data set is measured by the percentage of the minority side, which is binned into

three ranges: $[0.2, 0.3)$, $[0.3, 0.4)$ and $[0.4, 0.5)$.

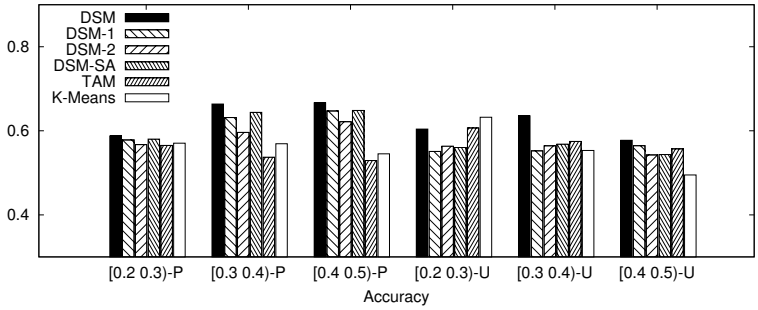

Figure 3.12: Comparisons of post side clustering ("-P") and user side clustering ("-U") accuracy in terms of data sets on different balance levels.

Results in Figure 3.12 shows that our model DSM clearly outperforms the baseline models including TAM and K-Means when the data sets are relatively balanced. For the unbalanced data set on user side clustering, K-Means shows better accuracy as we observe that K-Means tends to generate two unbalanced clusters, one with the majority points and the other with a few outlier points. K-Means also has relatively good performance on the unbalanced data set on post side clustering, but our model has better performance than it. In all, the average accuracy of DSM over all data sets on different balance levels is still the best, which shows the robustness of our model.

Impact of different types of features

We evaluate how our model performs on using different types of features in *split-tuple* representation as it shows better results than full-tuple representation.

Results are shown in Figure 3.13. We can make these observations: (1) The model results can be slightly improved by using N-gram features comparing to bag-of-word features. (2) Dependency features are proved to be important as adding which the model results are improved. (3) By adding negation features, the model results can be further improved comparing to adding dependency features. In terms of Accuracy, by adding negation features shows clear advantage by significantly outperforming other methods at 5% significance level measured by Wilcoxon signed

rank test. In all, by adding all three types of features, the model results can be significantly improved over the model with bag-of-words representation, at 1% significance level measured by Wilcoxon signed rank test.

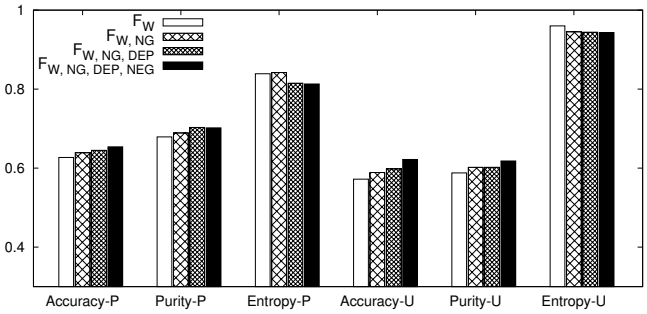

Figure 3.13: Impacts of different types of features on DSM in post side clustering ("-P") and user side clustering ("-U"). F_{W}, F_{NG}, F_{DEP}, and F_{NEG} stand for bag-of-words, N-gram, dependency relation and negation features respectively.

We have also studied adding polarity information to the opinionated features, the same as used in [74]. However, it does not improve the performance. One reason is that most of polarized features can be captured by the interaction model. We would like to emphasize that the language features studied in this work may be no way near all the language signals exhibit in user interactions, but rather a good set of language features that one can use to help the side clustering task in debates.

Generality of interaction features

Since our training data for learning interaction features and test data for user or post side cluttering are all from popular debates, it is thus possible to observe threads with similar topics in the training data with the test data. A natural question is whether the learned interaction features are only helpful to threads with similar topics. In other words, are our learned interaction features general enough for threads from other domains? To answer this question, we conduct the following experiment.

We asked an external examiner to evaluate whether the test data set has overlapping topics with the training data set (the data set used for finding interaction features). The study shows we have totally 11 debates out of 40 with topics similar

to those in the training data. Most of them are on the popular debate topics like "Does god exists?" (7 overlaps), "same-sex marriage"(2 overlaps) and "abortion"(2 overlaps). We then removed these overlapping debates and re-evaluated the results on the rest of the debates.

Method	Post Side Clustering			User Side Clustering		
	Accuracy	*Purity*	*Entropy*	*Accuracy*	*Purity*	*Entropy*
DSM	**0.653**‡	**0.708**†	**0.793**†	**0.592**‡	**0.615**	0.943$^{\diamond}$
DSM-1	0.624	0.690	0.820	0.567	0.596	0.969
DSM-2	0.620	0.651	0.888	0.580	0.601	0.955
DSM-SA	0.615	0.676	0.860	0.556	0.585	0.965
TAM	0.545	0.555	0.982	0.582	0.600	**0.941**$^{\diamond}$
Kmeans	0.559	0.561	0.976	0.578	0.596	0.961

Table 3.11: Comparisons of post side clustering and user side clustering results on data set without overlapping topics with training data set. ‡ means the result is better than others in the same column at 5% significance level measured by Wilcoxon signed rank test, † is at 10% level, and $^{\diamond}$ means the result is better than other methods without this symbol at 10% significance level.

We report the results in Table 3.11. In post side clustering, we find our model still significantly outperforms other baseline methods at 10% significant level. In user side clustering, we find our model still shows advantage over other models, but the differences between our model and other models are not as significant as the task of post side clustering. TAM shows a good performance on user side clustering, mainly because of the rich features used. Except for TAM, our model outperforms other methods at 10% significant level.

In summary, we actually observe similar findings where our model still shows better performance than other baseline models. This shows our model still has an advantage on training and test data sets with different topics. It suggests that the interaction features learned by our model are those general interaction features from different domains and may not be affected by domain shift.

Parameter Analysis - δ

Recall that parameter δ in our model serves as pseudo counts for the statement in the indicator function, as presented in Eqn. (3.8). In this section, we will study how this parameter affects the model performance. To simplify the comparison, we set $\delta_1 + \delta_0 = 1$. Note that, δ_1 and δ_0 represent to what extent we believe users from the same side tend to have positive interactions and users from different sides with negative interaction. If $\delta_1 < \delta_0$, we are inclined to have a stronger belief that users from different sides tend to have negative interactions.

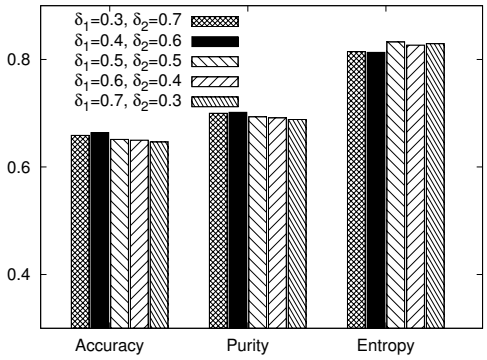

Figure 3.14: Impacts of δ on our model results in the post side clustering task.

Figure 3.14 compares our model for different ranges of values for δ in the post side clustering task. Although the performances measures are quite close, we observe that setting $\delta_1 < \delta_0$ yields relative better performance, which means we have more confidence in users with different sides having negative interactions than the same side having positive interactions. This correlates to our data set property, as we observe users with different sides almost always "dispute" to each other, while users with the same side do not always "support" or "clarify" each other. We find to model the interplay between interactions and sides is a challenging task. In the near future, we will apply more linguistic analysis to undercover the underlying relationship between them.

3.2.4 Summary

In this work, we propose an Interaction Model to uncover interaction features from structured debate posts with known sides and reply intentions such as those from CreateDebate. We then design our Debate Side Model to consider interaction features for debate side clustering. Empirical evaluation shows our DSM can perform significantly better for side clustering than the baseline models.

3.3 Discussion

In our first work, we proposed a novel latent variable model for viewpoint discovery from threaded forum posts. Our model is based on the three important factors: viewpoint specific topic preference, user identity and user interactions. Our proposed model captures these observations in a principled way. In particular, to incorporate the user interaction information, we proposed a novel generative process. Empirical evaluation on the real forum data sets showed that our model could cluster both posts and users with different viewpoints more accurately than the baseline models we consider. To the best of our knowledge, our work is the first to incorporate user interaction polarity into a generative model to discover viewpoints.

K-Means. Note that if we purely use bag-of-word representation, K-Means shows better performance than the Topic-Aspect Model (TAM) [74] in terms of post-level stance clustering as shown in Section 3.2.2. We test TAM and K-Means on the data sets used in Chapter 3 as well. We find that K-Means also shows better performance than TAM on the data set CDS2. A close examination shows that CDS2 and the data set used in Section 3.2.2 have high user interactions, where more than 80% of posts are reply posts. This shows that comparing to K-Means, TAM is more suitable for data sets with more user arguments related to the issue. For forum data sets, we observe a large amount of data sets are with user interactions, and by

modeling interaction features, our model has better performance than both TAM and K-Means.

Document representation. In this chapter, we shows that with more complex lexical units such as n-grams [65] and dependency triplets [74] can improve the performance of topic models. In our second work, we propose an Interaction Model to uncover interaction features from structured debate posts with known sides and reply intentions such as those from CreateDebate. We then design our Debate Side Model to consider interaction features and the interplay between interactions and sides for debate side clustering. Empirical evaluation shows our DSM can perform significantly better for side clustering than the baseline models.

In our data set, we observe some cases where users from the same side "dispute" with each others, which shows although two users may share the same side on a controversial topic, they may still disagree with each other on some factors. This relates to the controversy property of topics; some topics tend to be so controversial that users with the same side may not reach a good agreement. We would like to mine such controversy property of topics to help the side clustering tasks in the future.

Chapter 4

Modeling User Opinions

Advances in sentiment analysis have enabled extraction of user relations implied in online textual exchanges such as forum posts. However, recent studies in this direction only consider direct relation extraction from text. As user interactions can be sparse in online discussions, we propose to apply collaborative filtering through probabilistic matrix factorization to generalize and improve the opinion matrices extracted from forum posts. Experiments with two tasks show that the learned latent factor representation can give good performance on a relation polarity prediction task and improve the performance of a subgroup detection task.

4.1 Introduction

The fast growth of the social Web has led to a large amount of interest in online social network analysis. Most existing work on social network analysis relies on explicit links among users such as undirected friendship relations [50], directed following relations [37] and trust/distrust relations [48]. However, besides these explicit social relations, the various kinds of interactions between online users often suggest other implicit relations. In particular, in online discussion forums, users interact through textual posts and these exchanged texts often reveal whether two users are friends or foes, or whether two users share the same viewpoint towards a

given issue.

To uncover such implicit relations requires text analysis and particularly sentiment analysis. Recently, [35] studied predicting the polarity of user interactions in online discussions based on textual exchanges. They found that the automatically predicted signed relations had an accuracy above 80%. The extracted signed network was further used to detect ideological subgroups. This is a piece of pioneering work that extracts online social relations based on text analysis.

In this chapter, we further extend the idea of mining social relations from online forum posts by incorporating collaborative filtering. Our work is motivated by the observation that direct textual exchanges between users are sparse. For example, in the data set we use, only around 13% of user-user pairs have direct interactions. Collaborative filtering is a commonly used technique in recommender systems to predict missing ratings. The key assumption is that if two people have the same opinion on an item A, they are likely to also have the same opinion on a different item B. In online discussion forums, users express their opinions about each other as well as the various aspects of the topic under discussion, but not every user comments on every aspect or every other user. Collaborative filtering allows us to identify users with the same opinion even if they have not directly interacted with each other or commented on any common aspect.

Our method starts with extracting opinions on users and topic aspects from online posts using sentiment analysis. The results are two matrices indicating the sentiment polarity scores between pairs of users and pairs of a user and an aspect. To incorporate collaborative filtering, we choose probabilistic matrix factorization (PMF) [87], a technique that has been successfully applied for collaborative filtering-based recommendation problems. PMF automatically discovers a low-rank representation for both users and items based on observed rating data. In our problem, the predicted sentiment polarity scores are treated as rating data, and the results of PMF are low-rank vectors representing each user in online discussions.

We evaluate our method on two tasks. The first is to predict the polarity of

interactions between two users not from their own textual exchanges but from their interactions with other users or comments on topic aspects. The second is to use the latent vectors to group users based on viewpoints. We find that the latent factor representation can produce good prediction results for the first task and improve the clustering results of the second task compared with a number of baselines, showing the effectiveness of collaborative filtering for mining social relations from online discussions.

4.2 Method Overview

In this section, we provide an overview of our method. We first introduce some concepts.

User: We use *user* to refer to a discussant in an online discussion. Each user has an online ID, which can be used by other users to refer to him/her in a post. Users are both opinion holders and opinion targets. For example, User 1 below expresses a negative opinion towards another user in the following snippet.

*User 1: Actually, I have to disagree with **you**.*

Aspect: We use *topic aspect* or *aspect* to refer to an opinion target that is related to the topic under discussion. For example, when debating about whether one should vote for Obama, people may express opinions on targets such as "President Obama" and "Republican party," as shown in the following snippets. These aspects are all related to Obama's presidential campaign. As we will explain later, the aspects we consider are named entities and frequent noun phrases.

*User 2: Americans should vote for **President Obama** because he picks good corporations as winners.*

*User 3: I simply point out how absolutely terrible the **Republican party** is.*

Polarity Score: A sentiment polarity score is a real number between 0 and 1, where 0 indicates a completely negative opinion and 1 indicates a completely positive

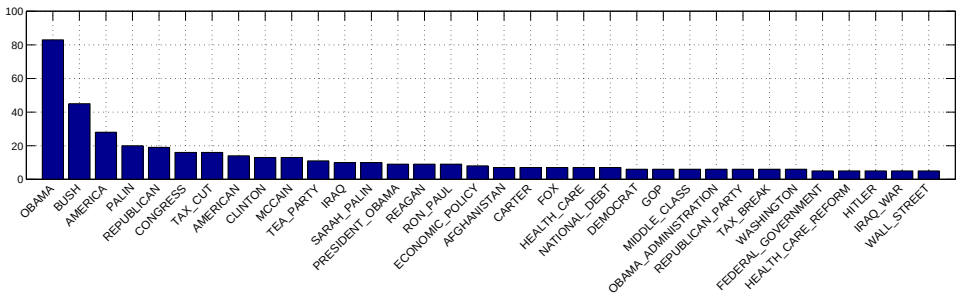

Figure 4.1: Salient aspects and number of users who express opinions on them in the thread "Will you vote for Obama?"

opinion.

User-User Opinion Matrix: The opinions extracted from posts between users are represented by a user-user opinion matrix S, where entry $s_{i,j}$ is a polarity score between the i-th user and the j-th user. We assume that the polarity scores are symmetric.

User-Aspect Opinion Matrix: The opinions held by different users on the various topic aspects are represented by a user-aspect opinion matrix R, where entry $r_{i,k}$ is a polarity score indicating the i-th user's opinion towards the k-th aspect.

Given the matrices S and R, we perform probabilistic matrix factorization to derive a low-rank vector representation for users and aspects such that if the polarity score between two users or a user and an aspect is high, the dot product between the corresponding two vectors is also high.

In Section 4.3, we will explain in detail how we identify topic aspects from a discussion thread and how we obtain polarity scores from posts. In Section 4.4, we will present the details of our probabilistic matrix factorization model.

4.3 Construction of Opinion Matrices

The opinion matrices are constructed from a single forum thread discussing some controversial topic.

4.3.1 Aspect Identification

As we have pointed out, there are two kinds of opinion targets, namely users and aspects. Users are clearly defined and can often be identified in posts by their IDs or second person pronouns. For aspects, however, there is not a pre-defined set. We observe that these topic aspects are usually named entities or noun phrases frequently mentioned. We therefore use the OpenNLP toolkit[1] to perform chunking and obtain noun phrases and the Standford NER tagger[2] to identify named entities from the posts.

Some of the candidate aspect phrases identified above actually refer to the same actual aspect, e.g. "Obama voter," "Obama voters" and "the Obama voter." We remove stop words from each candidate phrase and use the WordNet by [61] to obtain the lemma of each word such that we can normalize the candidate aspect phases to some extent.

Finally, to select salient aspects for a given discussion topic, we count the number of times each candidate aspect has been expressed a positive or negative opinion on by all users, and select those candidate aspects which have opinion expressions from at least M users. We set M to 2 in our experiments. Figure 4.1 shows the top salient aspects for the thread on "Will you vote for Obama?" We acknowledge there are still duplicate aspects in the results like "Republican Party" and "GOP". To normalize these aspects, some additional information such as Wikipedia entries and Google snippets may be considered. We will study this problem in our future work.

4.3.2 Opinion Expression Identification

Our next step is to identify candidate opinion expressions. This problem has been studied in [38], [78], and [36]. Based on previous work, we do the following. We first combine three popular sentiment lexicons to form a single sentiment lexicon:

[1] `http://opennlp.apache.org/`
[2] `http://nlp.stanford.edu/ner/index.shtml`

ID	Dependency path rule	Example
R1	$ADJ_{OP} \leftarrow amod \leftarrow N_{TR}$	I simply point out how *terrible* REPUBLICAN PARTY is.
R2	$ADJ_{OP} \rightarrow nsubj \rightarrow N_{TR}$	BUSH is even more *reasonable* for tax hike than Obama.
R3	$V_{OP} \rightarrow dobj \rightarrow N_{TR}$	I would never *support* OBAMA.
R4	$V_{OP} \rightarrow prep\_* \rightarrow N_{TR}$	I'll *vote* for OBAMA.
R5	$V_{OP} \rightarrow nsubjpass \rightarrow N_{TR}$	DEMOCRATIC PARTY are ultimately *corrupted* by love of money.
R6	$N_{OP} \leftarrow dobj \leftarrow V \rightarrow nsubj \rightarrow N_{TR}$	PAKISTAN is increasing terrorist *threat*.
R7	$ADJ_{OP} \leftarrow amod \leftarrow N \rightarrow nsubj \rightarrow N_{TR}$	OBAMA was a *top* scorer for occidental college.
R8	$ADV_{OP} \leftarrow advmod \leftarrow V \rightarrow nsubj \rightarrow N_{TR}$	OBAMA is *smarter* than people.

Table 4.1: Examples of frequent dependency path rules in our training data. OP and TR refer to the opinion and the target. The opinion words are in italic and the aspect words are in uppercase.

the lexicon used in [38], MPQA Subjectivity Lexicon by [102] and SentiWordNet by [8]. Our final sentiment lexicon contains 15,322 negative expressions and 10,144 positive expressions. We then identify candidate opinion expressions by searching for occurrences of words in this lexicon in the posts.

4.3.3 Opinion Relation Extraction

Given a post that contains an aspect and an opinion expression, we still need to determine whether the opinion expression is used to describe the aspect. This is a relation extraction problem. We use a supervised learning approach based on dependency paths. Previous work by [63], and [79] has shown that the shortest path between a candidate opinion aspect and a candidate opinion expression in the dependency parse tree can be effective in extracting opinion relations. We use the Stanford Parser from [44] to obtain the dependency parse trees for each sentence in the posts and then get the dependency paths between each pair of candidate aspect and opinion expression. We use dependency relations and POS tags of nodes along the path to represent a dependency path. Given a set of training sentences (we use the one from [103]), we can get a set of dependency path rules based on their frequencies in the training data. Table 4.1 shows the frequent dependency path rules in our training data.

When a pair of aspect and opinion expression is identified to be related, we use the polarity of the opinion expression to label the relation. Finally, given a

pair of users, we use the percentage of positive interactions between them over all subjective interactions (i.e. interactions with either positive or negative opinions) as extracted from their exchanged posts as the sentiment polarity score between the two users, regardless of the reply-to direction of the posts. Similarly, given a user and an aspect, we also use the percentage of positive opinion relations extracted as the sentiment polarity score between them. Thus the user-user opinion matrix and the user-aspect opinion matrix are constructed. If there is no subjective interaction detected between two users or between a user and an aspect, the corresponding entry in the matrix is left empty. We will see later that empty entries in the matrices are not used in the probabilistic matrix factorization step.

4.4 Probabilistic Matrix Factorization

As we have pointed out earlier, a problem with the matrices extracted as described in Section 4.3 is that the matrices are sparse, i.e. many entries are empty. For the data set we use, we find that around 87% of entries in the user-user opinion matrix and around 90% of entries in the user-aspect opinion matrix are empty. In this section, we describe how we use Probabilistic Matrix Factorization (PMF) to represent users and aspects in a latent factor space and thus generalize the user preferences.

Our model is almost a direct application of probabilistic matrix factorization from [87], originally proposed for recommender systems. The main difference is that the user-user opinion polarity scores are symmetric. Our model is also similar to the one used by [55]. We describe our model as follows.

We assume that there are K latent factors with which both users and aspects can be represented. Let $u_i \in \mathbb{R}^K$ denote the vector in the latent factor space for the i-th user, and a_k the vector for the k-th aspect.

Recall that the opinions extracted from posts between users are represented by a user-user opinion matrix S, and the opinions held by different users on the various topic aspects are represented by a user-aspect opinion matrix R. We assume that the

polarity scores $s_{i,j}$ between the i-th and the j-th users and $r_{i,k}$ between the i-th user and the k-th aspect in the two matrices S and R are generated in the following way:

$$\begin{aligned} p(s_{i,j}|u_i, u_j, \sigma_1^2) &= \mathcal{N}(s_{i,j}|g(u_i^T u_j), \sigma_1^2), \\ p(r_{i,k}|u_i, a_k, \sigma_2^2) &= \mathcal{N}(r_{i,k}|g(u_i^T a_k), \sigma_2^2), \end{aligned}$$

where σ_1^2 and σ_2^2 are variance parameters, $g(\cdot)$ the logistic function, and $\mathcal{N}(\cdot|\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2.

We can see that with this generative assumption, if two users are similar in terms of their dot product in the latent factor space, then they are more likely to have positive interactions as extracted from their textual exchanges. Similarly, if a user and an aspect are similar, then the user is more likely to express a positive opinion on the aspect in his/her posts. The latent factors can therefore encode user preferences and similarity between two users in the latent factor space reflects whether they share similar viewpoints.

We also place the following prior over u_i and a_k:

$$\begin{aligned} p(u_i|\sigma_U^2) &= \mathcal{N}(u_i|\vec{0}, \sigma_U^2\mathbf{I}), \\ p(a_k|\sigma_A^2) &= \mathcal{N}(a_k|\vec{0}, \sigma_A^2\mathbf{I}), \end{aligned}$$

where σ_U^2 and σ_A^2 are two variance parameters for users and aspects, respectively, and $\mathbf{I}$ is the identify matrix.

Figure 4.2 shows the plate notation for the generative model.

Let $\mathcal{U}$ be a $K \times U$ matrix containing the vectors u_i for all U users, and $\mathcal{A}$ be an $K \times A$ matrix containing the vectors a_k for all A aspects. To automatically learn $\mathcal{U}$

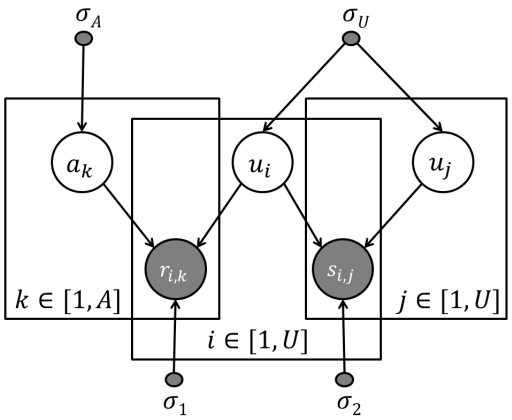

Figure 4.2: Probabilistic matrix factorization model on opinion matrices.

and $\mathcal{A}$, we minimize the following objective function:

$$\begin{aligned} & \mathcal{L}(\mathcal{U}, \mathcal{A}, \mathcal{S}, \mathcal{R}) \\ = \; & \frac{1}{2} \sum_{i=1}^{U} \sum_{k=1}^{A} \mathbb{I}(r_{i,k})(r_{i,k} - g(u_i^T a_k))^2 \\ + \; & \frac{\lambda_1}{2} \sum_{i=1}^{U} \sum_{j=1}^{U} \mathbb{I}(s_{i,j})(s_{i,j} - g(u_i^T u_j))^2 \\ + \; & \frac{\lambda_U}{2} ||\mathcal{U}||_F^2 + \frac{\lambda_A}{2} ||\mathcal{A}||_F^2, \end{aligned} \tag{4.1}$$

where $\lambda = \frac{\sigma_1^2}{\sigma_2^2}$, $\lambda_U = \frac{\sigma_1^2}{\sigma_U^2}$, and $\lambda_A = \frac{\sigma_1^2}{\sigma_A^2}$, $\mathbb{I}(s)$ is an indicator function which equals 1 when s is not empty and otherwise 0.

To optimize the objective function above, we can perform gradient descent on $\mathcal{U}$ and $\mathcal{A}$ to find a local optimum point. The derivation is similar to [55].

Degenerate Versions of the Model

We refer to the complete model described above as PMF-UOM (PMF model based on User Opinion Matrices). PMF-UOM has the following two degenerate versions by considering either only the user-user opinion matrix or only the user-aspect opinion matrix.

PMF-UU: In this degenerate version of the model, we use only the user-user opinion matrix to learn the latent factor representation. Specifically, the objective function is modified such that we drop the sum of the square errors involving R and the

regularizer on $\mathcal{A}$.

PMF-UA: In this degenerate version of the model, we use only the user-aspect opinion matrix to learn the latent factor representation. Specifically, the objective function is modified such that we drop the sum of the square errors involving S.

4.5 Experiments

In this section, we present our experiments that evaluate our model.

4.5.1 Data Set and Experiment Settings

The data set we use comes from [80], [2] and [35]. The data set contains a set of discussion threads collected from two political forums (Createdebate[3] and Politicalforum[4]) and one Wikipedia discussion session. Some details of the data we use are listed in Table 4.2. Note that although we use the same data sets from [80], the experiment results are not comparable as our work here only focus on the user opinion matrices extracted from the opinionated texts while the work in [80] also uses non-opinionated texts.

ID	topic	#sides	#sentences	#users
EDS1	Vote for Obama	2	12492	197
EDS2	Arizona Immigration Law	2	2500	59
EDS3	Tax Cuts	2	1193	26
EDS4	Abortion Banned	6	3844	70
EDS5	Profile Muslims	4	2167	69
EDS6	England and USA	6	2030	62
EDS7	Political Spectrum	7	1130	50

Table 4.2: Some statistics of the data sets.

In our experiments, for the PMF-based methods, we set the number of latent factors to be 10 as we do not observe big difference when varying the latent factor size from 10 to 50. For the other parameters, we select the optimal setting for each

[3] www.createdebate.com

[4] www.politicalforum.com

thread based on the average of 50 runs. λ_U is chosen from $\{0.1, 0.01\}$, λ_A from $\{0.01, 0.001\}$ and λ from $\{1, 0.1\}$.

4.5.2 Relation Polarity Prediction

The first task we use to evaluate our model is to predict the polarity of interactions between two users. Different from [35], however, we are not using this task to evaluate the accuracy of sentiment analysis from text. Our experimental setting is completely different in that we do not make use of the text exchanges between the two users but instead use their interactions with other users or aspects. The purpose is to test the effectiveness of collaborative filtering.

Experimental Setting: The experiments are set up in the following way. Given a pair of users i and j who have directly exchanged posts, i.e. $s_{i,j}$ is not empty, we first hide the value of $s_{i,j}$ in the matrix S. Let the altered matrix be $S_{\neg(i,j)}$. We then use $S_{\neg(i,j)}$ instead of S in the learning process as described in Section 4.4 to learn the latent factor representation. Let $\hat{u}_i$ and $\hat{u}_j$ denote the learned latent vectors for user i and user j. We predict the polarity of relation between i and j as follows:

$$\hat{s}_{i,j} = \begin{cases} 1 & \text{if } g(\hat{u}_i^T \hat{u}_j) > 0.5, \\ 0 & \text{otherwise}, \end{cases}$$

where $g(\cdot)$ is the logistic function to convert the dot product into a value between 0 and 1.

To judge the quality of the predicted polarity $\hat{s}_{i,j}$, we could compare it with $s_{i,j}$. But since $s_{i,j}$ itself is predicted from the textual exchanges between i and j, it is not the ground truth. Instead, we ask two human annotators to assign the true polarity label for user i and user j by reading the textual exchanges between them and judging whether they are friends or foes in the discussion thread. The annotators are asked to assign a score of 0 (indicating a negative relation), 0.5 (indicating a neutral relation) or 1 (indicating a positive relation). The lowest agreement score

based on Cohen's kappa coefficient among the 6 threads we use is 0.56, showing fair to good agreement. As ground truth, we set the final polarity score to 1 if the average score of the two annotators is larger than 0.5 and 0 otherwise.

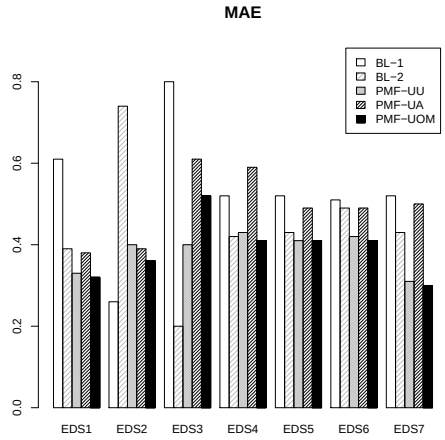

Figure 4.3: Comparing all the methods in terms of MAE.

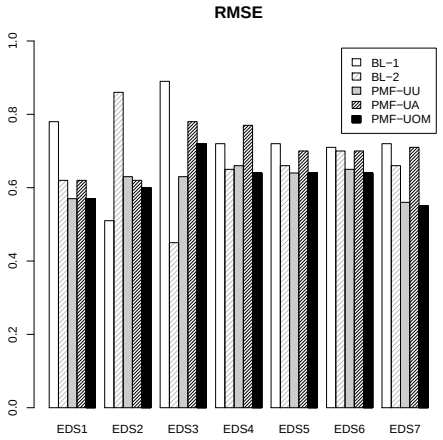

Figure 4.4: Comparing all the methods in terms of RMSE.

We compare the PMF-based methods with two majority baselines: MBL-0 always predicts negative relations for all the user pairs (assuming most relations are negative) and MBL-1 always predicts positive relations (assuming most relations are positive).

We use *MAE* (mean absolute error) and *RMSE* (root mean square error) as defined below as performance metrics:

$$\begin{aligned} \textit{MAE} &= \frac{\sum_{i,j} |\hat{s}_{i,j} - l_{i,j}|}{N}, \\ \textit{RMSE} &= \sqrt{\frac{\sum_{i,j} (\hat{s}_{i,j} - l_{i,j})^2}{N}}, \end{aligned}$$

where N is the total number of user pairs we test, and $l_{i,j}$ is the ground truth polarity score between user i and user j.

Results: We show the results of our model and of PMF-UU and PMF-UA in terms of MAE in Figure 4.3 and RMSE in Figure 4.4. The *MAE* values range between 0.31 and 0.44 except for EDS3, which has a higher error rate of 0.53. The results show that even without knowing the textual exchanges between two users, from

their interactions with other users and/or with topic aspects, we can still infer the polarity of their relation with decent accuracy most of the time.

The results also show the comparison between our model and the competing methods. We can see that overall the complete model (PMF-UOM) performs better than the two degenerate models (PMF-UU and PMF-UA). The differences are statistically significant at the 5% level without considering EDS3, as indicated by a 2-tailed paired t-test. Comparing to the majority baselines, our model significantly outperforms MBL-1 at 1% significance level while outperforms MBL-0 on all the data sets except EDS3. A close examinations shows EDS3 has very unbalanced relations (around 83% of relations are negative). Except for the unbalanced data set, our model has reasonably good performance.

4.5.3 Subgroup Detection

The second task we study is the problem of detecting ideological subgroups from discussion threads. The original data set has been labeled with the ground truth for this problem, that is, for each thread the number of viewpoints is known and the viewpoint held by each user is labeled. A subgroup is defined as a set of users holding the same viewpoint.

Experimental Setting: Through this second experiment, we would like to verify the hypothesis that using the learned latent factor representation $\mathcal{U}$ for users, we can better detect subgroups than directly using the opinion matrices S and R. For all the methods we compare, we first construct a feature vector representation for each user. We then apply K-means clustering to group users. The number of clusters is set to be the true number of viewpoints for each thread. The different methods are described below:

PMF-based methods: We simply use the learned latent vectors $\hat{u}_i$ after optimizing the objective function as the feature vectors to represent each user.

BL-1: This is our own implementation to simulate the method by [2]. Here

each user is represented by a $(3 \times (U + A))$-dimensional vector, where U is the number of users and A is the number of aspects, i.e. $(U + A)$ is the total number of opinion targets. For each opinion target, there are 3 dimensions in the feature vector, corresponding to the number of positive, neutral and negative opinion expressions towards the target from the online posts.

BL-2: BL-2 is similar to BL-1 except that we only use a $(U + A)$-dimensional vector to represent each user. Here for each opinion target, we directly use the corresponding sentiment polarity score $s_{i,j}$ or $r_{i,j}$ from the matrix S or R. For empty entries in S and R, we use a score of 0.5.

We use *Purity* (the higher the better), *Entropy* (the lower the better) to evaluate the performance of subgroup detection [56]. We further use *Accuracy* obtained by choosing the best alignment of clusters with the ground truth class labels and computing the percentage of users that are "classified" correctly.

Results: We first give an overview of the performance of all the methods on the task. We show the average performance of the methods on all the data sets in Figure 4.5. Overall, our model has a better performance than all the competing methods.

We present all the results in Figure 4.6, 4.7, and 4.8. We perform 2-tailed paired t-test on the results. We find the PMF-UOM model outperforms its degenerative models PMF-UU and PMF-UA at 10% significance level in terms of all the measures.

We observe that PMF-UOM achieves the best performance in terms of all the measures for almost all threads. In particular, comparison with BL-1 and BL-2 shows that collaborative filtering can generalize the user preferences and help better group the users based on their viewpoints. The fact that PMF-UOM outperforms both PMF-UU and PMF-UA shows that it is important to consider both user-user interactions and user-aspect interactions.

The Effects of Cluster Size: To test the effect of the number of clusters on the experiment result, we vary the number of clusters from 2 to 10 in all methods. We find that all methods tend to achieve better results when the number of clusters

equals the ground truth cluster size. Overall, our method PMF-UOM shows a better performance than the other four methods when the number of clusters changes, which indicates the robustness of our method.

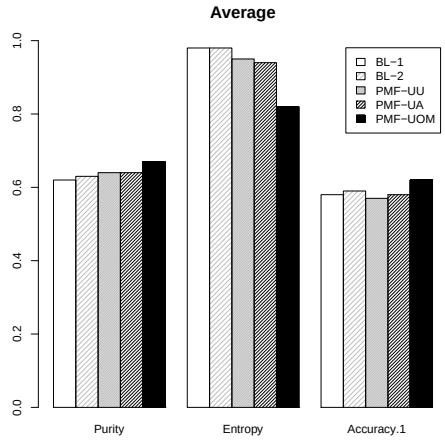

Figure 4.5: An overview of the averaged performance.

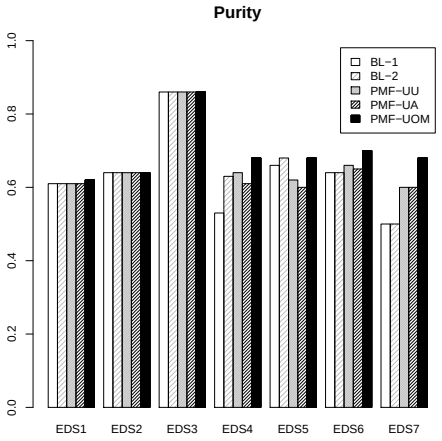

Figure 4.6: Comparing all the methods in terms of purity.

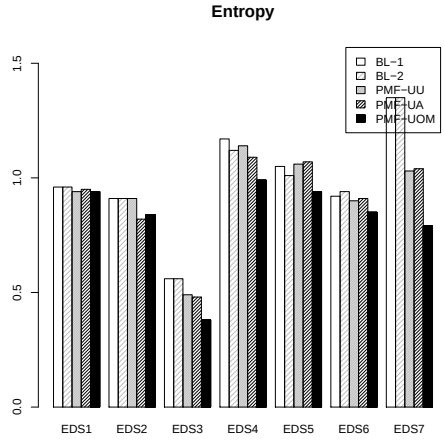

Figure 4.7: Comparing all the methods in terms of entropy.

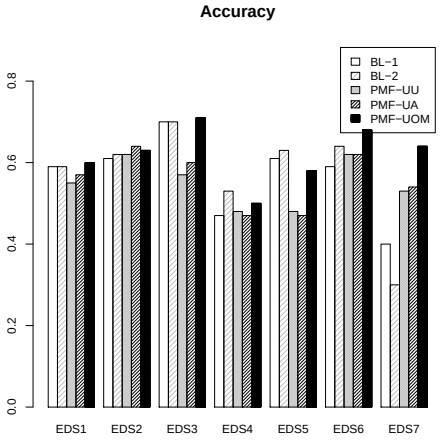

Figure 4.8: Comparing all the methods in terms of accuracy.

Our results show: (i). BL-1 and BL-2 are strong baselines as they perform comparable to PMF-UU and PMF-UA. The fact that PMF-UOM outperforms BL-1 and BL-2 show the effectiveness of using collaborative filtering way to find user groups; (ii). PMF-UA shows good performance on the data sets, this means by looking at users' opinions towards aspects can help to find user groups; (iii). PMF-UA have similar performance with PMF-UU shows that user aspect interaction have similar

importance in this task; (vi). We find PMF-UOM shows a robust performance on all the data sets, and it significantly outperforms PMF-UA and PMF-UU. This shows the effectiveness of combining both user user interaction and user aspect interaction for finding user groups.

4.6 Discussion

In this chapter, we studied how to use probabilistic matrix factorization, a common technique for collaborative filtering, to improve relation mining from online discussion forums. We first applied sentiment analysis to extract user-user opinions and user-aspect opinions from forum posts. The extracted opinions form two opinion matrices. We then applied probabilistic matrix factorization using these two matrices to discover a low-rank latent factor space which aims to better generalize the users' underlying preferences and indicate user similarities based on their viewpoints. Using a data set with 6 discussion threads, we showed that the learned latent vectors can be used to predict the polarity of user relations well without using the users' direct interaction data, demonstrating the effectiveness of collaborative filtering. We further found that for the task of subgroup detection, the latent vectors gave better performance than using the directly extracted opinion data, again showing that collaborative filtering through probabilistic matrix factorization can help address the sparseness problem in the extracted opinion matrices and help improve relation mining.

As future work, we seek to explore these tasks.

Differentiation between discriminate and general items. In stance prediction, it is possible to differentiate between discriminate and general items. Take opinion-targets as example, we find some opinion-targets are discriminate while others are general. For example, in our data set EDS1, support-Obama and against-Obama users tend to have different opinions to opinion-targets like Democrat Party, GOP, and healthcare act., but share similar opinions toward tea party or green party. Using

our model, we can predict all users' opinions on these opinion-targets and further divide these opinion-targets into discriminate and general ones. To better understand those discriminate opinion-targets can help us to further understand the contrastive opinions held among different subgroups.

Incorporation of general textual contents. This chapter focuses on mining user opinions. As future work, we would like to explore how to incorporate textual contents without opinionated expressions. One possible way is to consider "framing" in user arguments as studied in [80, 29, 89]. Another possible way is to consider the combination of matrix factorization and topic modeling as studied by [99] where we can use topic modeling to study textual contents.

Chapter 5

Micro-level and Macro-level Stance Prediction

Online debate forums are important social media for people to voice their opinions and debate with each other. Mining user stances or viewpoints from these forums has been a popular research topic. However, most current work does not address an important problem: for a specific issue, there may not be many users participating and expressing their opinions. The reason may be the issue is not interested to users or the issue is a new issue. Despite the sparsity of user stances, users may provide rich side information, for example, users may write arguments to back up their stances, interact with each other, and provide biographical information. In this work, we propose an integrated model to leverage side information. Our proposed method is a regression-based latent factor model which jointly models user arguments, interactions, and attributes. Our method can perform stance prediction for both warm-start and cold-start users. We demonstrate in experiments that our method has promising results on micro-level stance prediction. Our empirical study shows that our model has a good result on macro-level stance prediction, which shows the potential to complement traditional surveys and polls.

5.1 Introduction

Online debate forums are important social media for people to voice their opinions and engage in debates with each other. Mining user stances and viewpoints from these forums has been a popular research area [53, 92, 93]. One potential application is understanding public opinion; e.g., what are the popular stances on the Affordable Care Act, how do they associate with different subpopulations, and how are they changing over time? However, there may be a low online participation rate of Internet users in online discussion forums relating to any particular debate. For example, in our dataset collected from the debating website CreateDebate,[1] where users can explicitly state their stances on user-created debate topics, 278 people participated in the debate titled "Should guns be banned in America" while only 3 people participated in the debate "ObamaCare." As a result, if we consider all the registered users and existing debates on CreateDebate, only 0.4% of the full set of user stances are observed in the data. In this chapter, we are interested in *predicting* a user's stance on a debate topic in which she has not participated.

A target user may not be interested in an issue, but it is still important to predict her stance on the given issue. The reason is that for an issue such as one that may affect a government's or a company's decision making, it is beneficial to predict the public's opinion on it. And by gathering all the individual users' stances on an issue will give a macro-level stance prediction result for an issue. We can model the task as an item rating prediction task, where using a user's purchase history, her preference for a new item is to be predicted. Collaborative filtering [28] is a technique commonly used to alleviate the data sparsity problem in rating prediction. Probabilistic matrix factorization (PMF) [87] has been shown to be an effective collaborative filtering algorithm, and we extend PMF in our work.

One notable problem with PMF is "cold-start", i.e., it cannot perform stance prediction for users without any past stances. To alleviate this problem, we incor-

[1] `http://www.createdebate.com`

porate user attributes and user-generated content in online discussions. To be more specific, we incorporate three types of information that are prevalent in online discussions: user arguments that are used to back up their stances, user interactions from texts exchanged between users, and user attributes from their biographical information. Such rich information can help with stance prediction on both warm-start users and cold-start users and provide valuable user and issue profiling. Hence, we propose an integrated model that is capable of modeling such information, which can provide insights into user behaviors and issues.

Using data from CreateDebate, we propose a unified model for the task of user stance prediction. Firstly, to incorporate user attributes from their biographies, we use a regression based latent factorization method [4] to profile users. In this method, each user's latent factors are aggregated from factors associated with the user's attributes and user specific deviations. This setting allows us to profile users who have no past stances, i.e., we can do stance prediction on cold-start users. We further introduce a novel Binomial Matrix Factorization (BMF) model in the context of categorical ratings (i.e., user stances). This method extends the original PMF [87] model, which is designed for numerical ratings. Furthermore, users write arguments to support their stances, which provide textual cues to understand different topics involved in the issue. Like [57], we associate each latent factor dimension with a topic so as to produce an intuitive explanation for the hidden factors from BMF and improve stance prediction results. In addition, we find that incorporating features about a user's interaction network provides us with a way to infer relationships between users, which we can leverage to better predict user stances. To infer the model parameters, we consider Monte Carlo EM [97, 100] and adapt a fast inference method based on SparseLDA [106].

This work also aims to contribute to the problem of inferring public opinion from freely available social media text and metadata [68, 72, 95]. Such approaches have the potential to *complement traditional surveys and polls*. We focus on debate forums with rich user-contributed texts, opinions, and interactions on diverse

topics. We formulate the task around predicting held-out user stances. Although online forums have been explored in the past for questions such as predicting user stances [93, 92], detecting subgroups in online communities [2, 35, 14, 54], identifying user interactions [6, 19, 65, 64], and knowledge discovery [29], to the best of our knowledge, this is the first study on stance prediction in a debate forum leveraging rich user interaction metadata. We find our methods tend to agree with polls from Gallup, despite the fact that aggregated user stances are different from Gallup. This is promising as the polls are based on survey data collected through a labor-intensive procedure, while our method could serve as a cheap and effective way to complement them.

In summary, our contributions are as follows:

- We propose a regression based latent factor model which jointly models user arguments, interactions and attributes for stance prediction.
- We study a fast inference method for the model.
- Our experiments show promising results on micro-level stance prediction for both warm-start and cold-start users.
- Our experiments show that our model has a good result on macro-level stance prediction, which shows the potential to complement traditional surveys and polls.

5.2 Problem Definition

In Table 5.1, we present an excerpt of user arguments from a debate page in CreateDebate. In CreateDebate, each debate issue i focuses on a particular debate question, for example, *"Does God exist?"* Each debate issue has defined stances, which are usually "Yes" and "No" stances for the issue. In addition, each issue i has a set of threaded arguments, where each argument can be an independent post or a reply to an earlier argument. Each argument is authored by a user u, and explicitly

contains his stance $s_{u,i}$ on the particular issue, e.g., user A in our example takes the "Yes" stance. One user can write multiple arguments on an issue. We represent the text of the nth argument from user u on the issue i using a bag of bigrams $w_{u,i,n}$.[2] If the nth argument by user u is a reply/interaction post, the user u needs to specify whether she wants to "dispute," "support," or "clarify" the recipient post. We take advantage of this metadata using an interaction polarity $l_{u,i,n} \in \{\text{positive, negative}\}$, and assign $l_{u,i,n}$ to be "negative" when the user's argument disputes an earlier post and "positive" otherwise.

Debate: ***Does God Exist?***	
"Yes" stance	"No" stance
User A: Theists: I believe God exists. Atheists: I believe God doesn't exist. Both rely on belief . . . (*"Yes" stance*) ↻User B (*Dispute*): Whoops. wrong. more like "I don't believe in God." Both rely on belief. Thing about theists is they look beyond what they can see.. . . (*"No" stance*) ↻User A (*Dispute*): . . . Athiesm relies on belief. You BELIEVE there's no God . . . (*"Yes" stance*)	User J: If there is no evidence leading up to a God, I dont believe. . . (*"No" stance*) ↻User K (*Disputed*): . . . how can we prove that it doesn't exist??? We have no choice but to accept it (*"Yes" stance*) ↻User L (*Disputed*): So how can you argue for something that you cannot even interact with on a comparable level? (*"No" stance*)

Table 5.1: Sample arguments on the debate "Does God Exist?"

We crawled all arguments of two-sided debates from all 14 categories in the CreateDebate website.[3] For all the participants, we also collect these types of attributes from their biographical information: *party* (e.g., republican, democrat), *religion* (e.g., catholic, christian), *gender* (e.g., male, female), *status* (e.g., single, married), *education* (e.g., in college, post grad.), and *country* (e.g., U.S., Singapore). We leave other attributes like "age" and user self-description in biography as further study. Table 5.2 displays some statistics on the CreateDebate corpus. We find user-stance and interaction information are sparse in our data.

Our task is to predict a given user u's stance on a target issue i when the user has not expressed his stance on that issue. We refer to this as micro-level stance

[2]In this work, we use only a bigram representation. Eschewing unigrams have been shown to provide for more human interpretable terms without hurting task performance [89].

[3]Website crawled in April 2013.

# users	4,994
# issues	1,727
# arguments	39,549 (average 23 per issue)
# unigram tokens	55,308
# unique bigrams	154,724 (after pruning)
# user stances	17,843 (0.21% density)
# interaction links	23,935 (0.12% density based on a symmetric matrix, excluding self-links)
# user attributes	party (1430), religion (926), gender (1430), status (1432), education (835), country (1432)

Table 5.2: Statistics of the dataset. Bigrams containing stop words or punctuations are removed during pruning.

prediction. In Section 5.5.3, we also consider macro-level stance prediction, where we estimate the percentage of users holding a certain stance for a particular issue i.

5.3 Model

We approach the problem using a probabilistic graphical model. The graphical representation of the model can be found in Figure 5.1.

The model is composed of four parts: *i.*) *user profiling*, which considers a regression based latent factorization method to incorporate user attributes for profiling users; *ii.*) *user stance*, which contains a binomial matrix factorization method for modeling categorical stance data; *iii.*) *user arguments*, which incorporate textual cues in threaded posts; and *iv.*) *user interaction*, which integrates the positive and negative interaction attributes between users.

5.3.1 User Profiling

Inspired by the work in [4, 108], we consider a regression based latent factorization method for profiling users. Let $f_u \in \mathbb{R}^{P\times 1}$ denote user u's attributes. We use a binary representation, where each dimension of f_u is set as 1 if the corresponding attribute is present in user u, and 0 otherwise. In this study, we consider user attributes in these categories: *party*, *religion*, *gender*, *status*, *education*, and *country*.

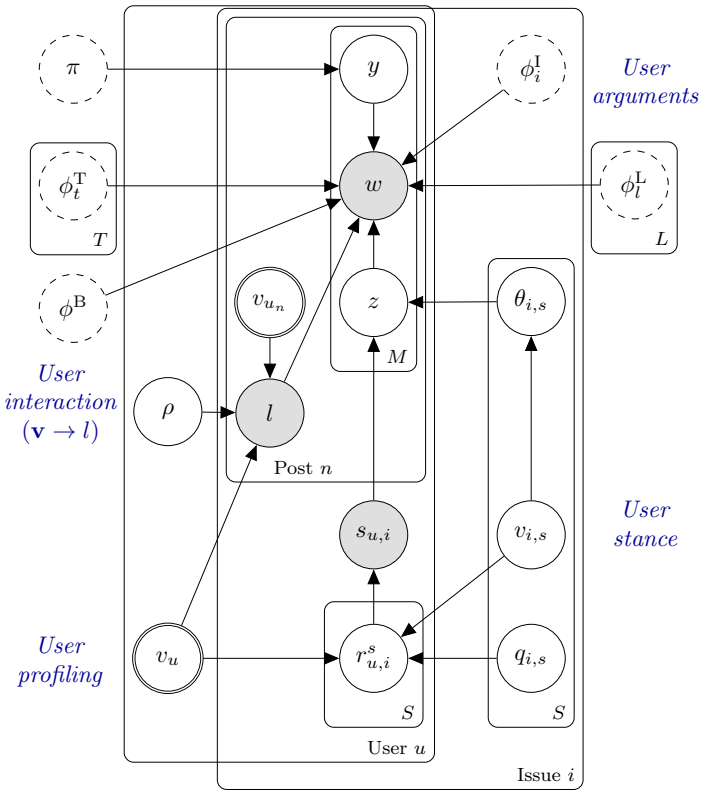

Figure 5.1: Plate notation for our model. The dashed variables will be collapsed out during Gibbs sampling. $\rho = \{c_1, c_2, q_u, q_{u_n}\}$, representing two parameters used in user interaction modeling and two biases specific to a user and her recipient. v_u, v_{u_n}, q_u and q_{u_n} are fixed by a regression based latent factorization method, detailed in Section 5.3.1. Hyperparameters are omitted for clarity.

We then model user latent factors v_u and bias q_u as:

$$\begin{aligned} v_u &= G^\top f_u + \delta_u, \\ q_u &= g^\top f_u + b_u, \end{aligned} \tag{5.1}$$

where $G \in \mathbb{R}^{P \times F}$ is a regression coefficient matrix, $g \in \mathbb{R}^{P \times 1}$ is a regression coefficient vector, $\delta_u \in \mathbb{R}^{F \times 1}$ is user specific deviation, and b_u is user specific bias. We pose zero-mean Gaussian priors on G, g, δ_u, and b_u.

5.3.2 User Stance

We assume that latent factor vectors of issues are drawn from zero-mean spherical Gaussian priors,

$$v_{i,s} \sim \mathcal{N}(0, \sigma_i^2 I),$$

where i and s refer to issue and stance, while hyperparameters σ_i^2 are issue-related variances. This differs from probabilistic matrix factorization in associating multiple factor vectors with a single issue. In this chapter, each issue corresponds to two vectors denoting the *support* and *oppose* stances; more stances could also be incorporated here.

Every user u has a rating on each stance s of an issue i,

$$r_{u,i}^s = v_u^\top v_{i,s} + q_{i,s}, \tag{5.2}$$

where $q_{i,s}$ is item-stance bias which is drawn from zero-mean Gaussian priors $\mathcal{N}(0, \sigma_q^2)$.

Using a logit function, the probability of user u choosing a stance s on issue i is

$$p(s_{u,i} = s) \propto \exp\{r_{u,i}^s\}. \tag{5.3}$$

We design such a setting to model the categorical rating data (stance) in debates. It captures the intuition that a user chooses a stance based on her own "ratings" over different stances. For example, in the debate "Do you support Obama or Romney in the presidential election?", a user tends to choose a stance "Obama" when her rating of "Obama" is higher than that of "Romney," i.e., $r_{u,i}^{\text{Obama}} > r_{u,i}^{\text{Romney}}$.

We refer this way of modeling user stance as binomial matrix factorization (BMF) as it extends probabilistic matrix factorization to two-sided stance data. This framework can be easily extend to multinomial model when some issues are with multiple sides.

5.3.3 User Arguments

We use a latent Dirchlet allocation topic model [10] to reduce the dimensionality of the text, and combine text data with latent factors from the user stance matrix, grounding each dimension of the hidden factor using inferred topics. Particularly, in our model, topics are in the same space as hidden factors, which is similar to the setting in [57].

We assume a stance-specific topic mixture $\theta_{i,s}$ for each stance s of an issue. The reason is that, for each issue, users with different stances tend to have different topic preferences [80]. $\theta_{i,s}$ denotes the relative log-odds of the different topics in issue i and stance s, encoding the distribution of topics that are likely to occur when arguing for that particular issue-stance. Specifically, $\theta_{i,s}^{\top} \propto \exp(v_{i,s}^{\top})$, where $v_{i,s}$ denotes the hidden factors associated with issue i and stance s.

The advantage of associating topic distributions with latent factors are two-fold. Firstly, the learnt topics provide an interpretation for factors, as each latent factor dimension is associated with a topic-specific word distribution. Secondly, this helps to reduce ambiguity for latent factors. In the BMF, v_u and $v_{i,s}$ can be replaced by Uv_u and $Uv_{i,s}$ if $U^{\top}U = 1$, as $v_u^{\top}v_{i,s} = (Uv_u)^{\top}(Uv_{i,s})$. This means, the factors may change considerably while leaving the underlying model unchanged. With the association, the θ learnt from texts will pose a regularization for the latent factors.

Moreover, we provide a fine-grained categorization of terms, where we assume the terms in a user's argument are drawn from one of the following four term distributions. Note that in this session, we use *term* to denote "bigram".

- Background term distribution ϕ^{B}. These are words uniformly distributed in many issues. For example, "united states," "no longer," and "things like."
- Issue-specific term distributions ϕ_i^{I}. Words that are related to the debate issue, e.g.: "God existence," "believe God" for the issue "Do you believe in God?"
- Topical term distributions ϕ_t^{T} for each topic t ($1 \leq t \leq T$). For example, terms like "health care," "federal government," and "tax cuts" are closely re-

lated to the topic "health care" and thus tend to have high probabilities under this topic.

- Interaction term distribution ϕ_l^{L} for each type of interaction l. These are words related to "positive" and "negative" interactions, for example, "i agree," "good point" for "positive" interaction, and "not agree," "not like" for "negative" interactions. In our work, the interaction polarity of an argument is observed, and this information is fed into our model to learn those interaction terms. Recent work in [65] also consider modeling interaction terms, however, they assume the interaction polarity is not available and use a Max-Ent component to guide the model to find those interaction terms.

Additionally, we incorporate switching variables y to decide from which term distribution a bigram is drawn [5, 73]. The generative story of our model on user arguments is

- Draw switching variable type distribution $\pi \sim \text{Dirichlet}(\gamma)$.
- Draw $\phi^{\mathsf{B}} \sim \text{Dirichlet}(\eta^{\mathsf{B}})$.
- $\forall$ interactions l, draw $\phi_l^{\mathsf{L}} \sim \text{Dirichlet}(\eta^{\mathsf{L}})$.
- $\forall$ topics t, draw $\phi_t^{\mathsf{T}} \sim \text{Dirichlet}(\eta^{\mathsf{T}})$.
- $\forall$ issues i, draw $\phi_i^{\mathsf{I}} \sim \text{Dirichlet}(\eta^{\mathsf{I}})$.
- $\forall$ stances s in issue i, the issue-stance topic distribution is determined by a logit function:

  $\theta_{i,s}^t \propto \exp(v_{i,s}^t)$.
- $\forall$ terms w in the m-th position of argument n from user u on issue i:
  - Draw switch $y_{u,i,n,m} \sim \text{Discrete}(\pi)$.
  - Draw topic $z_{u,i,n,m} \sim \text{Discrete}(\theta_{i,s})$.

- Draw term w,

$$w \sim \begin{cases} \text{Discrete}(\phi^{\mathsf{B}}) & \text{if } y = \mathsf{B} \\ \text{Discrete}(\phi^{\mathsf{I}}_i) & \text{if } y = \mathsf{I} \\ \text{Discrete}(\phi^{\mathsf{T}}_{z_{u,i,n,m}}) & \text{if } y = \mathsf{T} \\ \text{Discrete}(\phi^{\mathsf{L}}_{l_{u,i,n}}) & \text{if } y = \mathsf{L} \end{cases}$$

5.3.4 User Interaction

In CreateDebate data, user interactions are observed. As illustrated in Table 5.3, more positive user interactions are observed within users with the same stance and negative interactions in difference stances. We thus use it to inform our model in stance prediction, i.e., to guide the model to predict a user's stance to be the same with other users with who she has positive interactions, and different from those who she has negative interactions.

	Same stance	Different stance
Positive interactions	2,677 (71%)	1,101 (29%)
Negative interactions	932 (26%)	2,691 (74%)

Table 5.3: Confusion matrix for positive/negative interaction user pairs vs. user pairs with same/different stances. Interactions between users are aggregated across all in issues in our corpus.

This motivates us to associate the similarity of users in the latent factors with the polarity of user interactions. To measure the similarity of users in the latent factors, we simply use dot product of user factors and user biases. We leave other alternatives as future work. We then enforce a high probability of observing a positive interaction polarity for users who are highly similar. Specifically, let u' denote the recipient of user u's nth post in issue i, we sample user interaction polarity $l_{u,i,n}$ as:

$$\begin{aligned} p(l_{u,i,n} = +) &= \mathsf{S}\Big(c_1(v_u^\top v_{u'} + q_u + q_{u'}) - c_2\Big), \\ p(l_{u,i,n} = -) &= 1 - p(l_{u,i,n} = +), \end{aligned} \tag{5.4}$$

where $\mathsf{S}(\cdot)$ is logistic function, $c_1 \sim \mathcal{N}(1, \sigma^2)$ which encourages a positive value and

c_2 is also sampled from a Gaussian distribution with zero mean, i.e., $c_2 \sim \mathcal{N}(0, \sigma^2)$; q_u and $q_{u'}$ are user-specific biases, sampled from zero-mean Gaussian distribution $\mathcal{N}(0, \sigma_q^2)$; $v_u, v_{u'}, q_u$, and $q_{u'}$ are fixed by Eqn. 5.1.

5.4 Inference and Learning

Our goal is to learn the hidden factor vectors and topics of the textual content to accurately model user stances and maximize the probability of generating the textual content. Hence our objective function is defined as:

$$\begin{aligned} J = \quad & -\sum_{u,i,n} \Bigg(\log p(r_{u,i} \mid \rho_{u,i,n}) + \log p(l_{u,i,n} \mid \rho_{u,i,n}) \\ & + \log p(\rho_{u,i,n} \mid \Upsilon) + \log p(w_{u,i,n} \mid l_{u,i,n}, v_{i,s}, \Omega) \Bigg), \end{aligned}$$

where u, i, n are user, issue and argument index respectively. $\rho_{u,i,n} = \{v_i, q_i, G, g, \delta_u, \delta_{u'}, b_u, b_{u'}, c_1, c_2\}$ refers to the set of latent variables related to user u, recipient u' of the nth post of user u, and issue i, and Υ is the set of Gaussian priors for all the variables in $\rho_{u,i,n}$. Ω denotes all the Dirichlet prior hyperparameters for ϕ. The first three terms denote the probability of generating user stance and interaction given the priors Υ, where the variable in $\rho_{u,i,n}$ are to be optimized to minimize the objective function. The last term denotes the probability of observing the text conditioned on $\theta_{i,s}$ from learnt vector $v_{i,s}$, interaction $l_{u,i}$, and Dirichlet priors Ω.

Exact inference under the posterior distribution is intractable. We use Monte Carlo EM [97], an inference method that alternates between collapsed Gibbs sampling and gradient descent, to estimate parameters in the model. In the E-Step, we perform Gibbs sampling for variables $\{y, z\}$, fixing the values of ρ. In the M-step, we perform gradient descent to update latent variables in ρ, fixing the values of $\{y, z\}$.

5.4.1 E-Step.

We present the derived Gibbs sampling update rules and assume the reader is familiar with the approach. Interested readers are referred to [33] for more details.

For the term in the mth position of argument n from user u on issue i, we jointly sample its switching variable $y_{u,i,n,m}$ and topic $z_{u,i,n,m}$, conditioned on its Markov blanket. Let $w = w_{u,i,n,m}$, $s = s_{u,i}$ and $l = l_{u,i,n}$, let d denote the set of variables $\{u, i, n, m\}$.

$$
\begin{aligned}
&p(y_d = y, z_d = z | y_{\neg_d}, w, v_{i,s}, \Omega) \\
\propto &(C^{y}_{\neg_d} + \gamma) \cdot \left[\frac{C^{w}_{y,\neg_d} + \eta^{\mathsf{B}}_{w}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d} + V\eta^{\mathsf{B}}_{(\cdot)}} \frac{1}{T} \right]^{I(y=\mathsf{B})} \cdot \left[\frac{C^{w}_{y,\neg_d,i} + \eta^{\mathsf{I}}_{w}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d,i} + V\eta^{\mathsf{I}}_{(\cdot)}} \frac{1}{T} \right]^{I(y=\mathsf{I})} \\
&\cdot \left[\frac{C^{w}_{y,\neg_d,z} + \eta^{\mathsf{T}}_{w}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d,z} + V\eta^{\mathsf{T}}_{(\cdot)}} \theta^{z}_{i,s} \right]^{I(y=\mathsf{T})} \cdot \left[\frac{C^{w}_{y,\neg_d,l} + \eta^{\mathsf{L}}_{w}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d,l} + V\eta^{\mathsf{L}}_{(\cdot)}} \frac{1}{T} \right]^{I(y=\mathsf{L})}, \quad (5.5)
\end{aligned}
$$

where $C^{w}_{y=\mathsf{I},\neg_d,i}$ denotes the number of times that w is sampled as an issue-specific term in issue i excluding the current term assignment; all the other Cs are defined in the same way. $I(\cdot)$ is an indicator function. $\eta_{(\cdot)}$ is a summation over all the terms η_w. Note that, when $y = \mathsf{T}$, the term is a topical term, we need to sample a topic label from $\theta_{i,s}$, which is a deterministic logit transformation of $v_{i,s}$, specifically, $\theta^{z}_{i,s} = \frac{\exp(v^{z}_{i,s})}{\sum_t \exp(v^{t}_{i,s})}$.

5.4.2 M-Step.

In this step, we perform gradient descent to learn latent variables in ρ by fixing the values of y and z. We then reformulate the objective function.

$$\begin{aligned} J_{u,i,n} &= -\sum_{u,i,n} \Big(\log p(r_{u,i} \mid \rho_{u,i,n}) + \log p(l_{u,i,n} \mid \rho_{u,i,n}) + \\ &\quad \log p(\rho_{u,i,n} \mid \Upsilon) + \log p(\{z\}_{y=\mathsf{T}} | \theta_{i,s}, \Omega) \Big) \\ &= -\sum_{u,i,n} \Big(\log p(r_{u,i} \mid \rho_{u,i,n}) + \log p(l_{u,i,n} \mid \rho_{u,i,n}) + \\ &\quad \log p(\rho_{u,i,n} \mid \Upsilon) + \sum_{z} N^{z}_{u,i,n} \log \theta^{z}_{i,s} \Big) + O, \end{aligned} \tag{5.6}$$

where $N^{z}_{u,i,n}$ is the number of times topic z appears in user u's arguments in issue i. We used the expected counts obtained during the E-Step as we have assigned values to all the topics and switches. $\rho_{u,i,n}$ refers a set of latent variables $\{v_i, q_i, G, g, \delta_u, \delta_{u'}, b_u, b_{u'}, c_1, c_2\}$, and Υ is the set of Gaussian priors for all the variables in $\rho_{u,i,n}$. O is a constant that does not depend on the variables in $\rho_{u,i,n}$.

By computing first derivatives of J with respect to the variables in ρ, we can then update them using gradient descent.

5.4.3 Fast Inference.

Generally, the E-Step takes more time than the M-Step, since in the E-Step, we need to update topic and switch assignments for all the terms (bigrams). For each term, we jointly sample its corresponding topic z and switch y, and for $y = \mathsf{T}$, we sample topic from $\frac{C^{w}_{y,\neg_d,z} + \eta^{\mathsf{T}}_{w}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d,z} + V\eta^{\mathsf{T}}_{(\cdot)}} \theta^{z}_{i,s}$, otherwise we don't need to sample a topic label. Hence, each term takes $O(Y) + O(T)$ time to update, where Y is switch size, T is topic size.

To speed up the inference step, we consider the inference method used in SparseLDA [106]. In SparseLDA, it takes only $O(K_w + K_d)$ instead of $O(T)$ time to sample a topic for a word w in document d, K_w and K_d denote the number of

topics associated with w and d respectively. However, unlike SparseLDA, $\theta_{i,s}$ is a T-dimension dense term that cannot be further decomposed. Thus we resolve to use the following treatment.

$$\begin{aligned}
\frac{C^{w}_{y,\neg_d,z} + \eta^{\mathsf{T}}_{w}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d,z} + V\eta^{\mathsf{T}}_{(\cdot)}}\theta^{z}_{i,s} &= A(z) + B(z). \\
\text{where} \qquad A(z) &= \frac{C^{w}_{y,\neg_d,z}\theta^{z}_{i,s}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d,z} + V\eta^{\mathsf{T}}_{(\cdot)}}. \\
B(z) &= \frac{\eta^{\mathsf{T}}_{w}\theta^{z}_{i,s}}{\sum_{w'=0}^{V} C^{w'}_{y,\neg_d,z} + V\eta^{\mathsf{T}}_{(\cdot)}}.
\end{aligned}$$

Here $A(z)$ contains K_w elements, corresponding to the number of topics co-occurring with the term w, and $B(z)$ has T elements. To sample a topic, we first compute $\bar{A} = \sum_z A(z)$ and $\bar{B} = \sum_z B(z)$. We then choose $\bar{A}$ or $\bar{B}$ to proceed based on their proportions. With the data structure used in SparseLDA, and by storing encoded values of $(z, C^{w}_{y,\neg_d,z})$ in reverse-sorted arrays, we can calculate$\bar{A}$ and sample topic from $\bar{A}$ in $O(K_w)$ time. Note that $\bar{B}$ is the same for all the terms from issue i and stance s, that means to update $\bar{B}$ is cheap. As a result, with an initial cost for computing $\bar{B}$, it takes only $O(1)$ time to update $\bar{B}$ for a term. But to sample a topic from $\bar{B}$ takes $O(T)$ time. This means we only have a speed gain when we choose $\bar{A}$ to proceed.

In our experiment, we find $\frac{\bar{A}}{\bar{A}+\bar{B}} > 0.8$, which means, in most cases, we need only $O(K_w)$ to sample a topic. In all, to jointly sample a switch and a topic, for more than 80% of cases, we only need $O(Y) + O(K_w)$ time. We find this to be around three times as fast as the original method.

5.5 Experiments

Recall that our task is to predict a user's stance on an issue that she has not commented on. This problem setting is different from existing studies on stance prediction (e.g., [73, 92, 93]) where a user's arguments about an issue are observed

but not her stance, which makes the existing work not comparable. We then design experiments to: (*i.*) quantitatively evaluate our model with baselines on the tasks of micro-level stance prediction on warm-start and cold-start users. (*ii.*) examine our model on both macro-level stance prediction and compare it with polling data, (*iii.*) conduct an error analysis on the results, (*iv.*) analyze the efficiency of our inference method, and (*v.*) qualitatively examine term distribution of topics and issues learnt by our model.

5.5.1 Qualitative Analysis

We present six popular topics based on $\theta_{i,s}$ across issues in Table 5.4. Topic labels are manually assigned.

We find "existence of God" and "same-sex marriage" are popular topics in our data. All these topics are readily identified based on their top topical words. Topical terms are similar to high-level issues of the existence of God, healthcare, and same-sex marriage. Since these topics are in the same space as the hidden factors in matrix factorization, they can serve as interpretable labels for the corresponding dimensions in matrix factorization.

We present top interaction words for both positive and negative interactions from ϕ_l^{L} (see Table 5.5). These words are automatically learned by our model, making use of interaction polarity of user arguments. The results show these interaction words are quite intuitive. We also present top issue-specific terms from ϕ_i^{I} for popular issues in Table 5.5. These issues are hand picked by the authors from popular issues to cover a wider variety of issues as some issues are conceptually similar. Labels are assigned manually. Overall, these issue-specific terms the model discovers are easy to interpret. For example, on the issue "Does God Exist?", top terms are "no God", "scientific method," and "no proof." This shows that some users talk about the issue from a "science" perspective.

"Religion"	"Healthcare"	"Politics"	"Same-sex marriage"	"Death Penalty"	"Bin Laden"
god exists	health care	united states	gay marriage	death penalty	bin laden
no god	american people	barack obama	gay people	morally correct	al qaeda
prove god	federal government	white house	sexual orientation	life begins	al queda
christian god	tax cuts	bin laden	same-sex marriage	intense suffering	osama bin
richard dawkins	health insurance	foreign policy	equal rights	kill people	united states
atheists believe	wall street	democratic party	straight people	gay marriage	death penalty
agnostic atheist	social security	8 years	gay couple	chemical energy	no evidence
belief system	private sector	fox news	civil rights	moral agency	true true
against god	bush administration	republican party	gay couples	past tense	middle east
god told	small businesses	president obama	opposite sex	against israel	civilian casualties
no bearing	create jobs	president bush	sex marriage	earn money	human cost
lack belief	raise taxes	john mccain	against gay	no mind	iraq war
harry potter	economic crisis	sarah palin	consenting adults	equally bad	saudi arabia
evidence against	al qaeda	george bush	homosexual parents	evil equally	foreign policy
no faith	middle east	black people	gay rights	good number	military bases
modern science	higher taxes	osama bin	civil unions	thousand horsemen	vietnam war
jesus christ	voted against	ron paul	gay man	twelve thousand	armed forces
physical evidence	financial crisis	mitt romney	federal government	thousand stalls	political gain
god exist	track record	iraq war	gay sex	electrical energy	openly admit
believe god	billion dollars	bill clinton	born gay	muslim belief	million people

Table 5.4: Top topic terms from ϕ_t^{T}.

Negative	Positive	"Does God Exist?"	"Renewable Energy"	"Marines Urinating on Taliban"	"For/Against Gun Control?"
no evidence	good point	no god	wave energy	war crime	kill people
no god	health care	scientific method	energy technology	illegal invaded	balsthis sucks
no reason	no matter	natural sciences	offshore wave	war crimes	sucks balsthis
no matter	minimum length	no proof	total efficiency	official policy	ban guns
bin laden	totally agree	infinite religions	coal/nat gas	u.s. military	people kill
no longer	completely agree	jesus christ	sustainable energy	war logs	black market
al qaeda	god exists	no deity	onshore wave	accurate picture	gun control
no proof	years ago	natural laws	energy harnessing	invaded kuwait	banning guns
united states	looks like	religious law	good book	like torture	2nd amendment
people like	no reason	blame god	less feasilble	military operates	gun related
long time	good argument	morally perfect	viable comparison	dead bodies	gun laws
christian god	manhood academy	natural science	early 80s	u.s. armed	nuclear weapons
absolutely no	live debate	real science	due primarily	military code	related deaths
side supporting	dumb bitches	credible source	solar pv	civilian casualties	gun deaths
supporting mitt	common sense	vast majority	fuel mix	iraq invasion	guns illegal
al queda	look like	herd instinct	late 70s	like pissing	keep guns
sound like	human nature	infinite things	david mckay	little doubt	death rate
no idea	sounds like	immoral acts	cambridge david	dead human	less people
makes no	pretty good	great light	natural philosophy	u.s. army	save lives
sounds like	great point	infinite number	achieved efficiency	talk specifics	killing thousands

Table 5.5: Top interaction-specific terms from ϕ_l^{L}, and top issue-specific terms from ϕ_i^{I} for popular issues.

5.5.2 Micro-Level Stance Prediction

We conduct a study on micro-level stance prediction, i.e., predicting user stances on a given issue using learnt user and issue factor vectors and the user interaction network. We perform 10-fold cross-validation on our dataset. For each fold, we hold out as a test set 10% of the observed user-issue pairs, i.e., observed user stance on an issue. For each test set, if the issue does not appear in the training set, i.e., its a cold-start item, we will put it back to the training set. As for the rest test data, we split it into two datasets: one is for warm-start users, those users who have their past stances in the training set, the other for cold-start users, those who have no

past stances in the training set. Furthermore, we remove all the text associated with user-issue pairs in the test set and the prediction is based solely on users and issues factor vectors learnt from the training set. This setting mimics a real world scenario where a user does not have any prior stance on an issue, but the user has expressed stances on other issues and the issue has other users expressing their stances on it.

User Attributes.

We first examine the importance of different user attributes for stance prediction task. We use prediction accuracy (Acc) to measure model performance: Acc $= \frac{1}{|\mathcal{S}|}\sum_{u,i}\mathbb{I}(\hat{s}_{u,i} = s_{u,i})$.

We refer our base model without any user attributes as BMF-AI, binomial matrix factorization with user arguments and interactions. We evaluate the results by first considering only one type of attributes.

	BMF-AI	+ P	+ R	+ G	+ S	+ E	+ C
Acc	0.703	**0.707**	0.705	0.651	0.654	0.698	0.662
SD	0.008	0.008	0.007	0.007	0.008	0.007	0.008

Table 5.6: Micro-level stance prediction results on warm-start users by only incorporating one type of user attributes, averaged across ten folds. P, R, G, S, E, C stand for party, religion, gender, status, education, and country respectively. SD refers to standard deviations.

Table 5.6 shows that only these two types of attributes improve BMF-AI: party and religion. This shows that those attributes related to "ideology" are useful for the task of stance prediction. Furthermore, if we incorporate both party and religion attributes into the model, the results can be further improved to an accuracy of 0.712. In the following experiments, we will only incorporate these two attributes.

Warm-start Users.

We evaluate the following competing models for comparison.

- MB: majority baseline. For each test issue, we predict a user's stance based on the majority stance on the issue from the training data. This method performs

well when the stances are imbalanced, i.e., when an issue has a dominant stance.

- PMF: probabilistic matrix factorization [87]. The original model is designed for numerical ratings. We randomly map one stance of an issue to 0 and the other to 1.

- BMF: binomial matrix factorization. This model differs from PMF in that it assumes a rating for each stance of an issue and draws a stance based on a logit function over stance-specific ratings.

- HFT: hidden factors as topics [57]. Based on PMF, this model further considers user arguments.

- Our model and its variants: BMF-A: BMF with user arguments. BMF-AI: BMF with user arguments and interaction cues. BMF-AIA: BMF with user arguments, interactions, and attributes; it corresponds to the full model presented earlier.

	MB	PMF	BMF	HFT	BMF-A	BMF-AI	BMF-AIA
Acc	0.532	0.604$^\diamond$	0.607	0.642$^\diamond$	0.645	**0.703**$^\diamond$	**0.712**$^\diamond$
SD	0.015	0.012	0.012	0.011	0.012	0.008	**0.007**

Table 5.7: Micro-level stance prediction results, averaged across ten folds. $^\diamond$ The result is better than the method in the previous column at 5% significance level by McNemar's test. SD refers to standard deviations.

We find that (*i.*) MB performs poorly compared to other methods, suggesting that the stance data is fairly balanced, (*ii.*) both PMF and BMF significantly outperform MB, showing the collaborative filtering framework improves over the simple baseline, (*iii.*) BMF slightly outperforms PMF, implying that directly modeling of user-stance categories in BMF has a slight advantage over mapping the categorical ratings to numerical values as in PMF, (*iv.*) BMF-A significantly outperforms BMF, at 5% significance level, meaning that text is helpful in modeling user stances; with user arguments, we are able to bring together issues that are similar, as evidenced by

similar topic distributions. Meanwhile, BMF-A outperforms HFT by a small margin, this also shows the BMF is more general for the task, and (*v.*) modeling the user interactions can further boost performance, as shown by BMF-AI outperforming BMF-A, at 5% significance level, (*vi.*) by incorporating user attributes, the resulting model BMF-AIA achieves the best performance, significantly outperforms other competing methods. This demonstrates the effectiveness of an integrated model incorporating these information: user arguments, interactions, and attributes.

Cold-start Users.

"Cold-start" problem refers to predict new users' ratings on items which do not have any rating data; it is a common issue in recommendation systems. For a cold-start user, although we don't have her past stances, our model can still profile her using her attributes. Specifically, for a cold-start user u, we set its factor deviations $\delta_u = 0$, and $v_u = G^\top f_u$. The user's stance for a issue is from: $\arg\max_s \exp(v_u^\top v_{i,s})$. Here G and $v_{i,s}$ are learnt from training data.

	MB	BMF-AI		
		+ Party	+ Religion	+ Both
Acc	0.552	0.651	**0.662**	0.655
SD	n/a	0.028	0.029	**0.027**

Table 5.8: Micro-level stance prediction for cold-start users, averaged across ten folds. SD refers to standard deviations, n/a means not available.

We compare our method with different types of attributes and a majority baseline. The results are presented in Table 5.8. It shows that our model significantly outperforms the majority baseline, at 1% significance level by McNemar's test.

5.5.3 Macro-Level Stance Prediction

Recall that only 0.4% of the full set of user stances are observed in the data. We consider the task of predicting all the users' stances on all issues; the aggregate of these gives a macro-level stance prediction. Using our model, we can predict any

user's stance on any issue in our data giving all the learnt variables in ρ according to Eqn. 5.2 and Eqn. 5.1. Specifically, we set $\hat{s}_{u,i} = \arg\max_s \exp\{r^s_{u,i}\}$, and the macro stance for an issue i is defined as:

$$\hat{n}^s_i = \sum_u \mathbf{I}(\hat{s}_{u,i} = s). \tag{5.7}$$

For a select number of issues with existing Gallup poll data,[4] we can evaluate the proportion of the total number of users holding each stance $\frac{\hat{n}^s_i}{\sum_s \hat{n}^s_i}$ and compare them against the poll data.

High-level issues	# issues	# users	Majority stance proportions
Believe God	3	274	54% Yes
Same sex marriage	3	91	71% Support
Abortion	4	86	58% Pro-life
2012 election	2	48	55% For Obama
Gun control	3	317	67% Against
Obamacare	2	13	62% Against
Death penalty	3	61	63% In favor

Table 5.9: Stance proportions of CreateDebate high-level issues used for macro-level stance predictions. The number of users and the majority stance in the table is aggregated across all similar issues from known user stances in the data.

We find there are multiple issues that are phrased differently but arguably mean the same thing, e.g., "Does God exist?" and "Is there a God?" We refer a group of such similar issues as *high-level issues*. We chose seven high-level issues with the most arguments and have corresponding Gallup polls. We select Gallup polls whose (*i.*) poll date is closest to the CreateDebate data collection date[5] and (*ii.*) poll question is similar to all the issues in the high-level issue. For example, the high-level issue "gun control" contains three related issues: "Gun Control: Should we have it?", "Should guns be banned?", and "Should guns be banned in America?" and it corresponds to [25].[6]

[4] `http://www.gallup.com/`

[5] The CreateDebate website was crawled during the first week of April 2013.

[6] The corresponding Gallup polls for these high-level issues are: believe God [20], abortion [21], 2012 elections [22], Obamacare [23], death penalty [24], gun control [25], and same sex marriage [26].

For each issue i in CreateDebate, we know the stances of a small number of users and we can compute the proportions of users choosing the majority stance. We can also predict the proportions of the majority stance across the entire CreateDebate population with equation 5.7, and normalizing, $\frac{\hat{n}_{i,s}}{\hat{n}_{i,s}+\hat{n}_{i,\neg s}}$. Since we group similar issues together into high-level issues, stance proportions for similar issues are averaged to obtain stance proportions, which are presented in Table 5.9.

For these high-level issues, we identified Gallup poll results, from which we denote the ratios on both sides of the issue as $c_{i,s}$ and $c_{i,\neg s}$. In Gallup polls, users are allowed to provide "no opinion" as an answer, meaning we have $c_{i,s} + c_{i,\neg s} \neq 1$. We ignore this small subset of polled users, and instead normalize the ratios to get for stance s: $g_{i,s} = \frac{c_{i,s}}{c_{i,s}+c_{i,\neg s}}$.

High-level issues	Gallup	Prediction	Known
Believe God (Yes)	0.92	0.71	0.54
Same-sex marriage (Support)	0.51	0.60	0.71
2012 election (For Obama)	0.49	0.50	0.55
Abortion (Pro-life)	0.52	0.54	0.58
Gun control (Against)	0.75	0.69	0.67
Death penalty (In Favor)	0.63	0.65	0.64
Obamacare (Against)	0.53	0.64	0.61

Table 5.10: Comparison betweeen predicted and known proportions of users. "Prediction" refers to the predictions from our method, "Known" refers to the known proportions from CreateDebate stances.

The demographic of participants in CreateDebate may not be representative of the larger population surveyed by Gallup. Hence we expect that these stance estimates from CreateDebate do not match up with polling data from Gallup exactly. By comparing with polls from Gallup, we find our methods tend to agree with or bring the stance prediction results closer to those online polls. For example, the Gallup poll results for "2012 election," [21] has 49% of users vote for $Obama$. The CreateDebate corpus shows 55% of its users choosing against $Obama$. Our prediction model can leverage other correlations in data, and help to estimate that around 50% of users support $Obama$, which is closer to the Gallup poll. This suggests

that our methods can be used to complement traditional polling data collected via surveys. There are some interesting disagreeements, suggesting future directions improving the accuracy of text-measured public opinion and also text analysis to automatically characterize idiosyncratic opinions held in subcommunities.

5.5.4 Error analysis

(I.) Stance-specific biases

For our study on macro-level stance prediction, we find the high-level issues of "same sex marriage" and "belief God" have high prediction errors comparing to others. We consider these in greater depth.

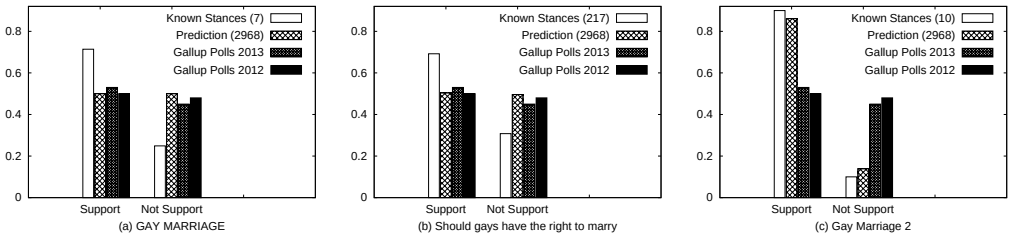

Figure 5.2: Macro-level stance prediction results on "Do You Support Gay Marriage?" (a) and (c) are two different issues but with the same title

Model	$q_i^{\text{“}Yes''}$	$q_i^{\text{“}No''}$	$q_i^{\text{“}Yes''} - q_i^{\text{“}No''}$
(a)	1.68×10^{-4}	-1.68×10^{-4}	3.36×10^{-4}
(b)	0.02	-0.02	0.04
(c)	0.07	-0.07	**0.14**

Table 5.11: Item specific biases for the issue "Do You Support Gay Marriage?"

For the high level issue "same sex marriage," we present results in individual issues in (Figure 5.2). A good agreement is observed in both Figure 5.2(a) and (b) but not in (c). One possible reason is that in (c), a relatively larger different between stance-specific biases exists, i.e., $|q_{i,s} - q_{i,s'}|$ for issue i is relatively larger than for other issues, as shown in Table 5.11.This large variance is due to the skewed user stances in this issue, 9 on supporting stance *versus* only 1 on the opposing side. This leads to the model explaining the data with larger bias on $q^{\text{“Yes”}}$ for the issue "Do you

support gay marriage", as exemplified in Table 5.11. With such biases, the model prediction of stances may not diverge from the known stances too much. For the Gallup poll results for "believe God," we have the same problem. The CreateDebate corpus has only 54% of its users choosing the "Yes" stance, while the Gallup poll titled "Do you believe in God?" has 92% of users taking a "Yes" stance. Our model predicts 71% which is closer to the Gallup results, but still it cannot diverge the results too much from known stances. If our model is accurate, it also suggests a dimension along which the population of CreateDebate users is quite different from Gallup's sample.

(II.) Representativeness of participants

CreateDebate reports that 89% of its users are between ages 20 and 40, 86.7% are from the United States, 85% are male, and 92% are single. These are based on self-reporting, but they give a strong sense that CreateDebate should not be taken as representative. Hence we expect that macro-level stance estimates do not match up with polling data from Gallup exactly. Moreover, no one—neither us nor Gallup—can be certain that these estimates are accurate, due to difficulties in measuring public opinion such as sampling biases, truthfulness of responses, the way questions are framed, etc. (Of course, experts like those at Gallup have invested a great deal in techniques for overcoming those challenges.) Nevertheless, understanding where and how the results diverge may give us a sense of how the CreateDebate population is different from the population polled by Gallup.

5.5.5 Efficiency

We test efficiency of our inference method with the original setting. Let Y denote the number of switch options, and T to be topic size. Recall that the original setting takes $O(Y) + O(T)$ to assign a topic and switch to a word, while our method takes only $O(Y)+O(K_w)$ at most of the cases. We run both settings using a machine with

2.5 GHz Intel Core i5 CPU, 8GB memory and 256GB SSD harddisk. We present the running time of both methods in Figure 5.3. We find our method is around three times as fast as the original setting.

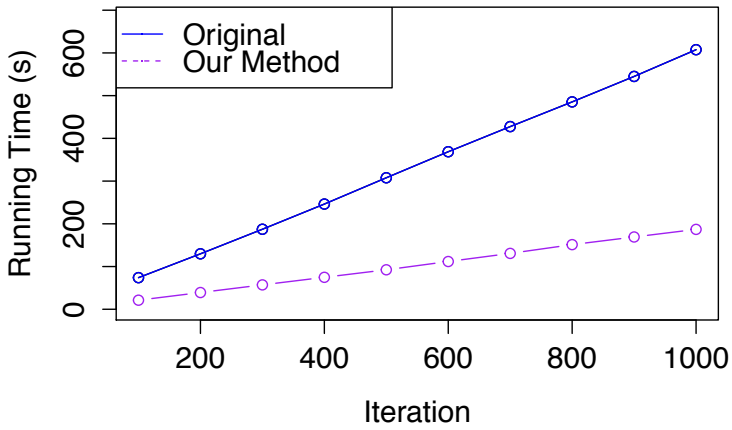

Figure 5.3: Running time of our fast inference method and the original setting.

Note that the efficiency of our inference method depends on the ratio of $\frac{\bar{A}}{\bar{A}+\bar{B}}$, as shown in Section 5.4. We plot this ratio in different iterations in Figure 5.4. We find, except for the first 200 iterations, the ratio is always larger than 86%. This shows our inference method has a time complexity of $O(Y) + O(K_w)$ in more than 86% of cases after 200 iterations.

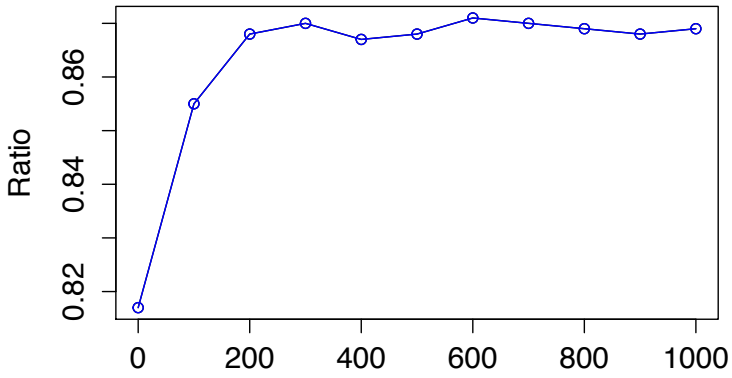

Figure 5.4: The averaged ratio of $\frac{\bar{A}}{\bar{A}+\bar{B}}$ from all the words in different iterations.

5.6 Discussion

In this work, we studied the novel setting of stance prediction task in the online discussion forum CreateDebate, with the goal of alleviating the data sparsity prob-

lem: there maybe a low online participation rate of Internet users in online discussion forums relating to any particular debate. We seek to predict user stances on a larger variety of topics and to complement traditional surveys and polls results. Our model brings together user arguments, interactions, and attributes into a probabilistic matrix factorization framework. To infer the model parameters, we considered a fast inference method based on SparseLDA. Experiments show promising results in micro-level stance prediction. Our empirical study also shows our model has a good result on macro-level stance prediction, which shows its potential to complement the traditional polls.

Limitations. We discuss a few limitations of the current model as below.

- User interactions. Firstly, there are cases where users have positive interactions on one issue, but negative interactions on another issue. This means it may be better if we profile user interaction polarity as a function of both user and issue factors. Furthermore, in our study, we find that people with the same viewpoint have negative interactions toward each other on some of the topics. For example, there is a case where two people believe in God, but one of them doesn't believe that "Jesus is God", so they will disagree with each other on this point. It would be beneficial to handle such inconsistency between user interactions and viewpoints.

- A diverse range of debates. Currently, we apply our collaborative framework on a diverse range of debates. We find that if we apply our method on just one domain, say "politics", the stance prediction results are better than the current results. This suggests that some information learnt from one domain may not be directly applied to other domain. For example, the model trained on the "education" domain may not be applicable to the "politics" domain. We are seeking ways to modify the current framework to adapt to different types of debates.

Future work. As future work, from a modeling perspective, a model with more

predictive power on unseen data may need to be considered. A simple extension may be to pursue a full Bayesian treatment of the probabilistic matrix factorization part. The idea is, instead of Gaussian priors, we assume Gaussian-Wishart priors on all the model variables related to matrix factorization. According to [87], this setting has higher predictive accuracy comparing to the original probabilistic matrix factorization. From a data perspective, the demographics of participants in CreateDebate may not be representative of the larger population. Hence we are interested in working on other social media such as Twitter and Facebook to extend our study.

Chapter 6

Viewpoint Summarization

In this chapter, we conduct an empirical study on viewpoint summarization. The task here is to find representative posts for each viewpoint of an issue, which can be viewed as viewpoint level extractive summarization. For such task, we can borrow techniques from multi-document summarization, which has been extensively studied in the NLP community, with most efforts on extractive summarization. Among all the summarization methods, Integer Linear Programming (ILP) based framework is a popular method. In this chapter, we consider this framework for user viewpoint summarization. We choose to build our solution based on ILP framework partially because in our preliminary analysis it outperforms other methods. Furthermore, it can be easily extended to incorporate more information. In our task, we make use of the information leant by our viewpoint discovery model discussed in Chapter 3, and based on which we score user posts by considering topic coverage and viewpoint distribution. We hypothesize that a good viewpoint specific summary should cover more topics and be more viewpoint specific.

In summary, our contributions are as follows:

- We extend the existing ILP based framework for the task of viewpoint summarization by leveraging the information leant by our viewpoint discovery model in Chapter 3.

- We evaluate the summarization method on real data sets by comparing with human generated reference summaries. Results show the proposed method has better performance than baseline approaches.

6.1 Task definition and method overview

We consider a set of forum posts published by different users on the same event or issue. For simplicity, we work on contrastive viewpoint data sets, i.e. data sets with two contrastive viewpoints. We focus on extractive summarization where we seek to find representative posts to form a summary for each viewpoint. As input, we assume that we have a set of posts separated into two viewpoints. For each viewpoint, the extractive summarization task is to select some representative posts within a given length limit of the summary.

Our method. Given the input as described above, we then use our proposed *JVTM-UI* model in Chapter 3.1 to automatically discover viewpoints. Note that by applying our model, we don't need to know user viewpoints beforehand. The model is based on three important factors: viewpoint specific topic preference, user identity and user interactions. Based on the model results, we seek to conduct a viewpoint specific summary by using the results produced by our JVTM-UI model. Specifically, we will make use of these two types of information: the relevance of selected posts with respect to a viewpoint and the coverage of viewpoint-specific topics. Below we present our method in details.

6.2 Model

In this section, we first briefly review ILP-based summarization framework and then present our proposed improvements.

The ILP framework was introduced as a global inference algorithm for multi-document summarization by [58], which considers informativeness and redundancy

at the sentence level. The work in [27] studies information and redundancy at "concept" level, modeling the value of a summary as a function of the concepts it covers. In their concept-based model, they use word bigrams weighted by the number of input documents in which they appear. The framework is recently used by many studies for multi-document summarization [34, 39, 49, 88, 104]. The idea behind the ILP framework by [27] is to maximize the coverage of so-called "concepts" from the original corpus in the generated summary. Specifically, assuming we have a set of posts from our data set, we use an index j to denote the j-th post. Let i be an index of all the concepts from the original corpus, a_i denote the weight of the i-th concept computed based on its frequency and $b_i \in \{0, 1\}$ denote the absence or presence of the concept, and L denote the length limit of the summary. The framework aims to maximize $\sum_i a_i b_i$, i.e. the total weighted coverage of the concepts, subject to a set of constraints:

$$\text{max:} \quad \sum_i a_i b_i \tag{6.1}$$

$$\text{s.t.:} \quad \sum_j l_j s_j \leq L, \tag{6.2}$$

$$\forall i, j : s_j o_{i,j} \leq b_i, \qquad \forall i : \sum_j s_j o_{i,j} \geq b_i. \tag{6.3}$$

where $s_j \in \{0, 1\}$ denotes the absence or presence of the j-th post, $o_{i,j} \in \{0, 1\}$ denotes whether concept i occurs in post j.

Constraint 6.2 ensures the summary is under the length limit. Constraint 6.3 ensures that b_i will be set to 1 when a post that contains the concept is selected, and 0 when no posts that contain the concept are selected.

Although this framework works well for standard summarization, our task is different as we need to provide a summary for each discovered viewpoint. We seek to provide a set of representative posts for each viewpoint by leveraging the output of JVTM-UI model. Hence we opt to consider the following constraints to form a summary.

Favoring viewpoint specific posts. The original ILP framework uses frequency

to measure the weight of a concept, but the most frequent bigrams in posts about a viewpoint may not be viewpoint specific as some of them may be related to the issue but not viewpoint specific. Ideally, we expect the selected posts to be more viewpoint specific. Hence we propose to use a score $q_{y,j}$ to measure the relevance of the j-th post with respect to a viewpoint y, and we use this score to select posts that are more relevant to viewpoint y. During pre-processing, a post with multiple sentences is separated into different posts where each post contains only one sentence. Recall that, we can learn the following variables by our model in Chapter 3.1: a viewpoint specific topic distribution θ^y for a viewpoint y, user-specific viewpoint distribution φ^u for each user u, and a topic-word distribution ψ_t for each topic t. We then compute $q_{y,j}$ as follows.

$$\begin{aligned} q_{y,j} &\propto p(y|u_j) \prod_{n=1}^{l_j} p(w_{j,n}|y) \\ &= \varphi_y^{u_j} \prod_{n=1}^{l_j} \sum_t p(w_{j,n}|t)p(t|y) \\ &= \varphi_y^{u_j} \prod_{n=1}^{l_j} \sum_t \psi_{w_{j,n}}^t \theta_t^y, \end{aligned} \tag{6.4}$$

where u_j is the author of post j, $\varphi_y^{u_j}$, and ψ_w^t and θ_t^y are learnt by our JVTM-UI model.

Here we use $\sum_j q_{y,j} s_j$ to denote the relevance between the selected posts and the viewpoint y. A larger $\sum_j q_{y,j} s_j$ means the selected posts are more relevant to the viewpoint y.

Covering viewpoint-specific topics: We hypothesize that a good viewpoint-specific summary should mention more viewpoint-specific topics. Since the topic label is at word level in our JVTM-UI model, we assign a topic label z_j for each post j by using this formula, $z_j = \arg\max_{t \in \mathcal{T}} \sum_y \varphi_y^{u_j} \theta_t^y \prod_{w \in \vec{w}_j} \psi_w^t$. Let $p_{j,k}$ denote whether topic t occurs in post j, i.e. $p_{j,t}$ is set as 1 when $t = z_j$ otherwise 0. Let $e_{y,t}$ denotes whether topic t is present in the selected posts for viewpoint y. We weight $e_{y,t}$ by θ_t^y to consider its relevance to the viewpoint y. Thus a larger $\sum_t \theta_t^y e_{y,t}$ means

the selected posts cover more viewpoint specific topics.

Eventually, for a viewpoint y, the viewpoint-specific summarization task is formulated as the following optimization problem:

$$\begin{aligned} \text{max:} \quad & \lambda_1 \textstyle\sum_i a_i b_i + \lambda_2 \sum_j q_{y,j} s_j + \lambda_3 \sum_t \theta_t^y e_{y,t} \\ \text{s.t.:} \quad & \textstyle\sum_j l_j s_j \leq L, \\ & \forall i : \textstyle\sum_j s_j o_{i,j} \geq b_i, \qquad \forall i, j : s_j o_{i,j} \leq b_i, \\ & \forall j : \textstyle\sum_t s_j p_{j,t} \geq e_{y,t}, \qquad \forall j, t : s_j p_{j,t} \leq e_{y,t}. \end{aligned} \tag{6.5}$$

where $\lambda_1 + \lambda_2 + \lambda_3 = 1$, $o_{i,j}$ denotes whether concept i occurs in post j, and $p_{j,t}$ denotes whether topic t occurs in post j.

We solve the above optimization problem using the IBM ILOG CPLEX Optimizer[1].

6.3 Experiments

6.3.1 Data and Experiment Setup

We need human generated summaries for evaluation. Since it is too time consuming to ask human annotators to look through all the posts and generate structured summaries, we instead opt to randomly choose a small set of our data to perform evaluation. To form our ground truth summaries, from all the posts related to a particular viewpoint, we randomly select 100 posts and present to two annotators. The annotators are asked to write a summary based on the given posts.

We use the following baseline algorithms for comparison: (1) **Random**, which randomly picks summary posts. (2) **ILP-BL**, which is the method proposed by [27]. (3) the degenerative versions of our proposed method: M-VP refers to our method without using viewpoint-specific posts, M-VT refers without using viewpoint-

[1] `http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/`

specific topics. For our method, with some preliminary analysis, we set $\lambda_1 = 0.3$, $\lambda_2 = 0.35$, and $\lambda_3 = 0.35$. We use ROUGE [51] scores as performance metrics.

6.3.2 Results

We first evaluate our method with the baseline methods by using the English data sets used in the Chapter 3. The results are evaluated against summaries from two annotators and then averaged as final results. Table 6.3.2 shows that our proposed method has better performance than other competing methods. The standard ILP method shows better performance than Random method. Our method has better performance when considering both viewpoint-specific posts and viewpoint-specific topics.

	EDS1		EDS2		EDS3	
Method	R-1	R-2	R-1	R-2	R-1	R-2
Random	0.30	0.09	0.19	0.03	0.31	0.08
ILP-BL	0.31	0.11	0.24	0.05	0.40	0.11
M-VT	0.32	0.13	0.26	0.06	0.44	0.13
M-VP	0.31	0.11	0.25	0.05	0.41	0.11
Our Method	**0.35**	**0.14**	**0.28**	**0.07**	**0.45**	**0.15**

Table 6.1: Comparison of the summarization results. M-VP refers to our method without using viewpoint-specific posts, M-VT refers without using viewpoint-specific topics.

	EDS1		EDS2		EDS3	
Method	R-1	R-2	R-1	R-2	R-1	R-2
Random	0.30	0.09	0.19	0.03	0.31	0.08
ILP-BL	0.31	0.11	0.24	0.05	0.40	0.11
Our Method	**0.35**	**0.14**	**0.28**	**0.07**	**0.45**	**0.15**

Table 6.2: Comparison of the summarization results. M-VP refers to our method without using viewpoint-specific posts, M-VT refers without using viewpoint-specific topics.

We show a sample summary our method generates for EDS1 in Table 6.4. We can see that all the posts selected by our method have high relevance to the corresponding viewpoints.

	EDS1		EDS2		EDS3	
Method	R-1	R-2	R-1	R-2	R-1	R-2
ILP-BL	0.31	0.11	0.24	0.05	0.40	0.11
M-VT	0.32	0.13	0.26	0.06	0.44	0.13
M-VP	0.31	0.11	0.25	0.05	0.41	0.11
Our Method	**0.35**	**0.14**	**0.28**	**0.07**	**0.45**	**0.15**

Table 6.3: Comparison of the summarization results. M-VP refers to our method without using viewpoint-specific posts, M-VT refers without using viewpoint-specific topics.

Do you support Obama?	
"Support" Obama	"Against" Obama
Obama has actually cut useless or wasteful government programs in order to save a few billion dollars here and there.	Obama is the living breathing embodiment of American leftism and we hate American leftism with our beings.
Obama has spent less than Bush did ...	Obama's 1st term increased the national debt 4x more than Bush's worst term.
Obama has frozen the federal budget for many agencys, created a national debt commision, cut some military spending, and turned the economy around from massive reccesion to small growth.	He sent troops to middel eastern war than bush did.
Sure we would all like more , but his policies are far better than republican ones ...	In his healthcare bill, he used backdoor meetings to secure votes by unions etc. ...
The number of illegal immigrants deported is at record highs. More troops are in Afghanistan and on the Border.	I don't think Democrats want to secure the border, they want those new potential Democrat voters.
Our allies like Obama a hell of a lit better than the last republican ...	They are raising taxes as we write and the fire they are putting under American butts is what is stemming the tide of continued wasteful spending.
	Democrats want to legalize the illegal immigrant , this is how they'll get the vote.

Table 6.4: Excerpts from the summary generated from EDS1 by our method.

6.4 Discussion

In this chapter, we proposed a summarization framework to find representative posts for different viewpoints for a controversial issue based on our proposed latent variable model discussed in Chapter 3.1 and integer linear programming (ILP). The latent variable model could align forum posts with different viewpoints for the controversial issue, and based on which the ILP model is used to distill representative posts for each discovered viewpoint. Comparing to standard ILP methods, our method additionally tries to cover more viewpoint-specific topics and viewpoint specific words. Experiments show our method outperforms other competing methods.

Chapter 7

Dissertation Conclusion and Future Work

To automatically discover and summarize user viewpoints from online discussions is important for both normal users and policy makers. In this dissertation, we study the task of mining user viewpoints in online discussions. We proposed principled approaches for these tasks in mining user viewpoints in online discussions, namely, viewpoint discovery, micro-level and macro-level stance prediction, and conducted an empirical study on user viewpoint summarization.

We start by two studies on modeling user posting behaviors for viewpoint discovery. Our first model takes three important characteristics of online discussions into consideration, namely, user consistency, topic preference, and user interactions. Our second model focuses on mining interactions features for the task of viewpoint discovery. Empirical evaluation shows our proposed models have significant improvement over other baseline methods. Furthermore, we study how to model user opinion matrices for viewpoint discovery. Our model makes use of the advances in sentiment analysis to extract user opinions in online user interactions, and feeds them into a collaborative filtering framework to profile users in a low-rank latent factor space. Experiments show the resulting low-rank representations of users makes

it feasible to cluster users by viewpoints. We also study micro-level and macro-level stance prediction. We propose an integrated model that jointly models user arguments, interactions, and attributes for stance prediction. Evaluation shows our model has promising results on both micro-level stance prediction and macro-level stance prediction. Last but not least, we study how to summarize user viewpoints. We consider extractive summarization to find representative user arguments to summarize viewpoints. We choose to build our solution on top of an integer linear programming based framework proposed by [27]. Experiments show by using results from our viewpoint discovery model, our method produces better summaries.

As future work, we consider a few directions to strengthen our studies as follows.

- Deep linguistic analysis. An immediate next step is to conduct deep linguistic analysis on how users frame their arguments for their viewpoints. Our work in [29] infers a low-dimensional, human-interpretable representation in the domain of issues and positions[1]. Another recent work shows that, with some known ideoogies, one may able to infer ideological cues from a corpus of political writings and using which to measure political candidates ideological positioning from their speeches. Inspired by these work, we seek to find cue terms that are associated with users' ideological behaviors and viewpoints to help mine user stances.

- Deep neural networks. The current advances in deep neural network provide a new way to model the semantic meanings of words, phrases and sentences [60, 76, 90]. It paves the way for the study of aspect based sentiment analysis and can be potentially used for the task of mining user viewpoints. Some recent works along this direction can be found in [16, 42, 47, 91, 101].

- Automatic detection of controversial topics. So far we have been worked

[1]This variable might serve to cluster debate sides according to "abstract" beliefs commonly shared by a group of people, sometimes called *ideologies*. We do not claim that our model infers ideologies.

on data sets that are known to be on controversial topics. In online discussions, many discussions are not controversial, hence to make the work more useful, it's important to automatic detect controversial issues from a general corpus. To achieve this goal, we need to mine user opinions on different opinion-targets in a given corpus, if we can find subgroups with high intra-group agreement and high inter-group disagreement, we can define the corpus as with controversial topics. We can then apply our developed models on the data. This is especially useful for social medias like Twitter and Facebook as in which rich user generated texts are observed but user viewpoints are implicit. To understand the controversiness and underlying stances of user arguments in these social medias can help to understand the public opinions and characterize idiosyncratic opinions held in subcommunities.

Furthermore, user opinions expressed in texts and social networks can help to profile users and provide better recommendation service. It's thus beneficial to jointly model texts, user interactions, and social network for user attribute discovery and recommendation systems. We have done a few studies along this direction.

- User attribute discovery. User demographic attributes such as gender, age, financial status, region are critically important for many business intelligence applications such as targeted marketing as well as social science research. Unfortunately, for reasons including privacy concerns, these pieces of user information are not always available from online social media platforms. Automatic discovery of such attributes from other observable user behavior online has therefore become an important research topic, which we call the user attribute discovery problem for short. In this work [31], we proposed an unsupervised integrated approach based on probabilistic matrix factorization that combines social and interactions features in a principled way to discover user demographics.

- Recommendation. Mining user interaction and opinion networks can be use-

ful for recommendation systems. Our first work extends the probabilistic matrix factorization method to incorporate user interaction network and user opinion network for mining user relations [83]. Our second work [15] proposes a recommendation model for jointly modeling aspects, ratings and sentiments for movie recommendation. Our models offer superior performance by joint modeling. Moreover, we are able to address the cold start problem by utilizing the information inherent in texts, user interactions or social networks. Our work in [85] also shows the importance of modeling user behaviors together with the generated texts.

Lastly, with the rapid growth of online social media, there are always new challenges and applications in user profiling and recommendation systems that require the advances of text mining, user opinion mining, and statistical machine learning.

Bibliography

[1] R. Abbott, M. Walker, P. Anand, J. E. Fox Tree, R. Bowmani, and J. King. How can you say such things?!?: Recognizing disagreement in informal political argument. In *Proceedings of the Workshop on Language in Social Media*, 2011.

[2] A. Abu-Jbara, P. Dasigi, M. Diab, and D. R. Radev. Subgroup detection in ideological discussions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, 2012.

[3] A. Abu-Jbara and D. R. Radev. Subgroup detector: A system for detecting subgroups in online discussions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics System Demonstrations*, 2012.

[4] D. Agarwal and B.-C. Chen. Regression-based latent factor models. In *Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2009.

[5] A. Ahmed and E. P. Xing. Staying Informed: Supervised and Semi-Supervised Multi-view Topical Analysis of Ideological Perspective. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2010.

[6] J. Andreas, S. Rosenthal, and K. McKeown. Annotating agreement and disagreement in threaded discussion. In *Proceedings of International Conference on Language Resources and Evaluation*, 2012.

[7] R. Awadallah, M. Ramanath, and G. Weikum. Polaricq: polarity classification of political quotations. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2012.

[8] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *The International Conference on Language Resources and Evaluation*, 2010.

[9] R. M. Bell and Y. Koren. Lessons from the Netflix Prize Challenge. *SIGKDD Explorations Newsletter*, 2007.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003.

[11] E. Cabrio and S. Villata. Combining textual entailment and argumentation theory for supporting online debates interactions. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics : Short Papers*, 2012.

[12] Y. Choi, E. Breck, and C. Cardie. Joint extraction of entities and relations for opinion recognition. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, 2006.

[13] M. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, A. Flammini, and F. Menczer. Political Polarization on Twitter. In *International AAAI Conference on Web and Social Media*, 2011.

[14] P. Dasigi, W. Guo, and M. T. Diab. Genre independent subgroup detection in online discussion threads: A study of implicit attitude using textual latent semantics. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2012.

[15] Q. Diao, M. Qiu, C. Wu, A. J. Smola, J. Jiang, and C. Wang. Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014.

[16] C. dos Santos and M. Gatti. Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*, 2014.

[17] G. Erkan and D. Radev. Lexpagerank: Prestige in multi-document text summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2004.

[18] Y. Fang, L. Si, N. Somasundaram, and Z. Yu. Mining contrastive opinions on political texts using cross-perspective topic model. In *Proceedings of the ACM international conference on Web search and data mining*, 2012.

[19] 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 *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2004.

[20] Gallup. Americans continue believe god, May 2011. http://www.gallup.com/poll/147887/Americans-Continue-Believe-God.aspx.

[21] Gallup. Americans split along pro-choice pro-life lines, May 2011. http://www.gallup.com/poll/147734/Americans-Split-Along-Pro-Choice-Pro-Life-Lines.aspx.

[22] Gallup. Romney obama gallup final election, November 2012. http://www.gallup.com/poll/158519/romney-obama-gallup-final-election-survey.aspx.

[23] Gallup. Americans say health-law harmful helpful, May 2013. http://www.gallup.com/poll/166793/americans-say-health-law-harmful-helpful.aspx.

[24] Gallup. Death penlty support lowest years, October 2013. http://www.gallup.com/poll/165626/death-penalty-support-lowest-years.aspx.

[25] Gallup. Remains divided passing stricter gun laws, November 2013. http://www.gallup.com/poll/165563/remains-divided-passing-stricter-gun-laws.aspx.

[26] Gallup. Sex marriage support solidifies above, May 2013. http://www.gallup.com/poll/162398/sex-marriage-support-solidifies-above.aspx.

[27] D. Gillick and B. Favre. A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, 2009.

[28] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry. Using Collaborative Filtering to Weave an Information Tapestry. *Communications of the ACM – Special issue on Information Filtering*, 1992.

[29] S. Gottipati, M. Qiu, Y. Sim, J. Jiang, and N. A. Smith. Learning Topics and Positions from Debatepedia. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2013.

[30] S. Gottipati, M. Qiu, L. Yang, F. Zhu, and J. Jiang. Predicting user's political party using ideological stances. In *Social Informatics - 5th International Conference, SocInfo 2013, Kyoto, Japan, November 25-27, 2013, Proceedings*, 2013.

[31] S. Gottipati, M. Qiu, L. Yang, F. Zhu, and J. Jiang. An integrated model for user attribute discovery: A case study on political affiliation identification. In *Advances in Knowledge Discovery and Data Mining - 18th Pacific-Asia Conference, PAKDD 2014*, 2014.

[32] S. Greene and P. Resnik. More than words: syntactic packaging and implicit sentiment. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2009.

[33] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 2004.

[34] A. Haghighi and L. Vanderwende. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2009.

[35] A. Hassan, A. Abu-Jbara, and D. Radev. Detecting subgroups in online discussions by modeling positive and negative relations among participants. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2012.

[36] A. Hassan and D. Radev. Identifying text polarity using random walks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2010.

[37] J. Hopcroft, T. Lou, and J. Tang. Who will follow you back?: reciprocal relationship prediction. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2011.

[38] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004.

[39] Y. Hu and X. Wan. Ppsgen: Learning to generate presentation slides for academic papers. In *Proceedings of the 21st international jont conference on Artifical intelligence*, 2013.

[40] M. Jamali and M. Ester. TrustWalker: A Random Walk Model for Combining Trust-based and Item-based Recommendation. In *The ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

[41] M. Jamali and M. Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of ACM Recommender Systems*, 2010.

[42] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao. Target-dependent twitter sentiment classification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2011.

[43] M. Joshi and C. Penstein-Rosé. Generalizing dependency features for opinion mining. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics : Short Papers*, 2009.

[44] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2003.

[45] Y. Koren and R. M. Bell. Advances in collaborative filtering. In *Recommender Systems Handbook*. 2011.

[46] L.-w. Ku, Y.-s. Lo, and H.-h. Chen. Using polarity scores of words for sentence-level opinion extraction. In *Proc. of the NTCIR-6 Workshop Meeting*, 2007.

[47] Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning*, 2014.

[48] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *Proceedings of the 19th international conference on World wide web*, 2010.

[49] C. Li, X. Qian, and Y. Liu. Using supervised bigram-based ilp for extractive summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2013.

[50] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2003.

[51] C. Lin and E. Hovy. From single to multi-document summarization: A prototype system and its evaluation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2002.

[52] W.-H. Lin, T. Wilson, J. Wiebe, and A. Hauptmann. Which side are you on?: Identifying perspectives at the document and sentence levels. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, 2006.

[53] W.-H. Lin, E. Xing, and A. Hauptmann. A joint topic and perspective model for ideological discourse. In *Machine Learning and Knowledge Discovery in Databases*. 2008.

[54] Y. Lu, H. Wang, C. Zhai, and D. Roth. Unsupervised discovery of opposing opinion networks from forum discussions. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2012.

[55] H. Ma, H. Yang, M. R. Lyu, and I. King. Sorec: Social recommendation using probabilistic matrix factorization. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2008.

[56] C. D. Manning, P. Raghavan, and H. Schtze. *Introduction to Information Retrieval*. Cambridge University Press, July 2008.

[57] J. McAuley and J. Leskovec. Hidden Factors and Hidden Topics: Understanding Rating Dimensions with Review Text. In *Proceedings of ACM Recommender Systems*, 2013.

[58] R. McDonald. A study of global inference algorithms in multi-document summarization. In *Proceedings of the 29th European Conference on IR Research*, 2007.

[59] Q. Mei, J. Guo, and D. Radev. Divrank: the interplay of prestige and diversity in information networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2010.

[60] T. Mikolov, Q. V. Le, and I. Sutskever. Exploiting similarities among languages for machine translation. *ACM Computing Research Repository*, 2013.

[61] G. A. Miller. Wordnet: A lexical database for english. *Communications of the ACM*, 1995.

[62] S. min Kim and E. Hovy. Crystal: Analyzing predictive opinions on the web. In *Proceedings of the joint Conference of the Conference on Empirical Methods in Natural Language Processing and Conference on Computational Natural Language Learning*, 2007.

[63] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing*, 2009.

[64] A. Mukherjee and B. Liu. Mining contentions from discussions and debates. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012.

[65] A. Mukherjee and B. Liu. Modeling review comments. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2012.

[66] C. C. Musat, Y. Liang, and B. Faltings. Recommendation using textual opinions. In *Proceedings of the international jont conference on Artifical intelligence*, 2013.

[67] A. Nenkova, L. Vanderwende, and K. McKeown. A compositional context sensitive multi-document summarizer: exploring the factors that influence summarization. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 2006.

[68] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From Tweets to Polls: Linking Text Sentiment to Public Opinion Time Series. In *International AAAI Conference on Weblogs and Social Media*, 2010.

[69] Y. Ouyang, W. Li, Q. Lu, and R. Zhang. A study on position information in document summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, 2010.

[70] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, 1999.

[71] B. Pang and L. Lee. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2008.

[72] M. J. Paul and M. Dredze. You Are What You Tweet: Analyzing Twitter for Public Health. In *International AAAI Conference on Weblogs and Social Media*, 2011.

[73] M. J. Paul and R. Girju. A two-dimensional topic-aspect model for discovering multi-faceted topics. In *The AAAI Conference on Artificial Intelligence*, 2010.

[74] M. J. Paul, C. Zhai, and R. Girju. Summarizing contrastive viewpoints in opinionated text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2010.

[75] Y. Pei, W. Yin, et al. Generic multi-document summarization using topic-oriented information. In *PRICAI 2012: Trends in Artificial Intelligence*. 2012.

[76] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2014.

[77] I. Persing and V. Ng. Vote prediction on comments in social polls. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2014.

[78] A.-M. Popescu and O. Etzioni. Extracting product features and opinions from reviews. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2005.

[79] G. Qiu, B. Liu, J. Bu, and C. Chen. Expanding domain sentiment lexicon through double propagation. In *Proceedings of the 21st international jont conference on Artifical intelligence*, 2009.

[80] M. Qiu and J. Jiang. A latent variable model for viewpoint discovery from threaded forum posts. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2013.

[81] M. Qiu, Y. Li, and J. Jiang. Query-oriented keyphrase extraction. In *Proceedings of Information Retrieval Technology, 8th Asia Information Retrieval Societies Conference,* , 2012.

[82] M. Qiu, Y. Sim, N. A. Smith, and J. Jiang. Modeling User Arguments, Interactions, and Attributes for Stance Prediction in Online Debate Forums. In *Proceedings of 2015 SIAM International Conference on Data Mining*, 2015.

[83] M. Qiu, L. Yang, and J. Jiang. Mining user relations from online discussions using sentiment analysis and probabilistic matrix factorization. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2013.

[84] M. Qiu, L. Yang, and J. Jiang. Modeling interaction features for debate side clustering. In *Proceedings of the ACM International Conference on Information and Knowledge Management*, 2013.

[85] M. Qiu, F. Zhu, and J. Jiang. It Is Not Just What We Say, But How We Say Them: LDA-based Behavior-Topic Model. In *Proceedings of 2013 SIAM International Conference on Data Mining*, 2013.

[86] D. Radev, H. Jing, M. Styś, and D. Tam. Centroid-based summarization of multiple documents. *Information Processing & Management*, 2004.

[87] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, 2008.

[88] C. Sauper and R. Barzilay. Automatically generating Wikipedia articles: a structure-aware approach. In *Proceedings of the Joint Conference of the Proceedings of the Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing*, 2009.

[89] Y. Sim, B. D. L. Acree, J. H. Gross, and N. A. Smith. Measuring Ideological Proportions in Political Speeches. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2013.

[90] R. Socher and C. D. Manning. Deep learning for nlp (without magic). The Association for Computational Linguistics, 2013.

[91] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013.

[92] S. Somasundaran and J. Wiebe. Recognizing stances in online debates. In *Proceedings of the Joint Conference of the Proceedings of the Annual Meeting of the Association for Computational Linguistics and the 4th International Conference on Natural Language Processing*, 2009.

[93] S. Somasundaran and J. Wiebe. Recognizing stances in ideological on-line debates. In *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010.

[94] X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advance in Artificial Intelligence*, 2009.

[95] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Election Forecasts With Twitter: How 140 Characters Reflect the Political Landscape. *Social Science Computer Review*, 2010.

[96] A. Tversky and D. Kahneman. The framing of decisions and the psychology of choice. *Science*, 1981.

[97] H. M. Wallach. Topic Modeling: Beyond Bag-of-words. In *Proceedings of The International Conference on Machine Learning*, 2006.

[98] X. Wan, J. Yang, and J. Xiao. Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of the 20th international joint conference on Artifical intelligence*, 2007.

[99] C. Wang and D. M. Blei. Collaborative topic modeling for recommending scientific articles. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011.

[100] G. C. Wei and M. A. Tanner. A monte carlo implementation of the em algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association*, 1990.

[101] R. West, H. S. Paskov, J. Leskovec, and C. Potts. Exploiting social network structure for person-to-person sentiment analysis. *Transactions of the Association for Computational Linguistics*, 2014.

[102] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2005.

[103] Y. Wu, Q. Zhang, X. Huang, and L. Wu. Phrase dependency parsing for opinion mining. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2009.

[104] L. Yang, J. Jiang, L. Huang, M. Qiu, and L. Liao. Generating supplementary travel guides from social media. In *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference*, 2014.

[105] S.-H. Yang, A. J. Smola, B. Long, H. Zha, and Y. Chang. Friend or Frenemy?: Predicting Signed Ties in Social Networks. In *Proceedings of the Annual International ACM SIGIR Conference*, 2012.

[106] L. Yao, D. M. Mimno, and A. McCallum. Efficient methods for topic model inference on streaming document collections. In *The ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

[107] S. Yardi and D. Boyd. Dynamic debates: An analysis of group polarization over time on twitter. *Bulletin of Science, Technology and Society*, 2010.

[108] L. Zhang, D. Agarwal, and B.-C. Chen. Generalizing matrix factorization through flexible regression priors. In *Proceedings of ACM Recommender Systems*, 2011.