Modeling Interaction Features for Debate Side Clustering

Minghui Qiu[†], Liu Yang[†,‡], Jing Jiang[†]
[†] School of Information Systems, Singapore Management University, Singapore
[‡] School of Software and Microelectronics, Peking University, China
[†] {minghui.qiu.2010,jingjiang}@smu.edu.sg, [‡] yang.liu@pku.edu.cn

ABSTRACT

Online discussion forums are popular social media platforms for users to express their opinions and discuss controversial issues with each other. To automatically identify the sides/stances of posts or users from textual content in forums is an important task to help mine online opinions. 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. This paper 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.

Categories and Subject Descriptors

I.2.7 [**ARTIFICIAL INTELLIGENCE**]: Natural Language Processing—*Language models, Text analysis*; H.3.1 [**INFORMATION STORAGE AND RETRIEVAL**]: Content Analysis and Indexing—*Linguistic processing*

Keywords

Side/Stance identification; Interaction features; Latent variable model

1. INTRODUCTION

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 paper 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 [9, 20]. 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[1], 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 1 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)`[2] 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 paper, 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 [18] and to infer user relations by looking at their textual exchanges [12]. These approaches require either sentiment lexicons, which may not be designed for user interactions, or labeled training data, which is labor-intensive to create. In the interaction feature identification

CIKM'13, Oct. 27–Nov. 1, 2013, San Francisco, CA, USA.
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-2263-8/13/10 ...$15.00.
http://dx.doi.org/10.1145/2505515.2505634.

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

[2] `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)
<u>User A</u>: Theists: I believe God exists. Atheists: I believe God doesn't exist. Both rely on belief . . . (*Side 0*) ↻<u>User B</u> (*Disputed*): Whoops. **wrong**. more like "I don't believe in god." . . . it is **gullibility** and **foolishness**. . . (*Side 1*) ↻<u>User A</u> (*Disputed*): . . . You BELIEVE there's no God. you cannot even prove it (*Side 0*) ↻<u>User C</u> (*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*)	<u>User J</u>: If there is no evidence leading up to a God, I dont believe. . . (*Side 1*) ↻<u>User K</u> (*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*) ↻<u>User L</u> (*Disputed*): So **how can you** argue for something that you cannot even interact with on a comparable level? (*Side 1*) ↻<u>User M</u> (*Supported*): Question:Why did the crusades happen? Answer: god told the people to kill muslims . . . (*Side 1*)

Table 1: Sample posts on the debate "Does God Exist?"

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.

2. 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 1, 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 $N \leq 3$. 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 [14], we thus consider adding syntactic features to our model. For each post, we use the Stanford parser [15] to get its dependency relations. For example, for the above sentence, we will get these relations:`nsubj(prove,you)`, `aux(prove,can)` and `neg(prove,not)`[3]. This representation is referred to as *full-tuple* representation. As this representation has low generalization power, *split-tuple* representation is used in [11, 14]. 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 [21]. 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.

2.1 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" [17, 26]. 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 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 1.

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[4]. 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 1 and the generative process is in Figure 2.

2.2 Interaction Features

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 [9] where the authors assume opinion words are adjectives, verbs and adverbs.

[3] `you cannot prove` will be tokenized as `you can not prove` by using the Stanford parser [15].

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

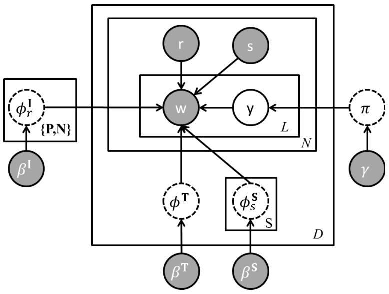

Figure 1: Interaction Model for modeling interaction words using the CreateDebate data. Dashed variables will be collapsed out in Gibbs Sampling.

- Draw selector distribution $\pi \sim$ Dir(γ)
- For each interaction type $r \in \{\mathsf{P}, \mathsf{N}\}$
  - Draw $\phi_r^{\mathsf{I}} \sim$ Dir(β^{I})
- For the d-th thread ($d = 1, 2, \cdots, D$)
  - Draw $\phi_d^{\mathsf{T}} \sim$ Dir(β^{T})
  - Draw $\phi_{d,s}^{\mathsf{S}} \sim$ Dir(β^{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 2: The generative process of the interaction model for CreateDebate. "Dir" and "Multi" stand for Dirichlet and Multinominal respectively.

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 the Stanford POS tagger [25]. Note that these are unigrams; (2) Words that appear in one of the following opinion lexicons: the sentiment lexicon used in [13], Multi-Perspective Question Answering Subjectivity Lexicon [27] and SentiWordNet [5]; (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 [18]; (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:

$$\phi_{r,w}^{\mathsf{I}} = \frac{C_{r,w}^{\mathsf{I}} + \beta^{\mathsf{I}}}{\sum_{w=1}^{V} C_{r,w}^{\mathsf{I}} + V\beta^{\mathsf{I}}}. \quad \text{interaction-word distr.} \quad (1)$$

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. 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. Before we formally present our model, we describe the main assumptions in the model.

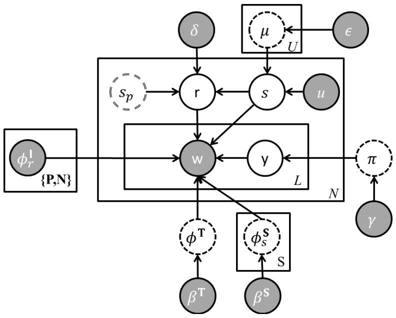

Figure 3: Plate notation for the Debate Side Model (DSM) on a given debate. Dashed variables will be collapsed out in Gibbs sampling. Double bordered dash variables are not new variables but a subset of the s variables.

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.

3.1 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 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 2. The plate notation of DSM is in Figure 3 and the generative process for the reply relations and the side information for the n-th post is shown in Figure 4.

- 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 $s_n \sim$ Multi(μ_{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 4: 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:

$$\begin{aligned} p(r_n = 1|s_n, s_n^{\mathsf{P}}, \boldsymbol{\delta}) &= \frac{\mathbb{I}(s_n^{\mathsf{P}} == s_n) + \delta_1}{1 + \delta_1 + \delta_0}, \quad (2) \\ p(r_n = 0|s_n, s_n^{\mathsf{P}}, \boldsymbol{\delta}) &= 1 - p(r_n = 1|s_n, s_n^{\mathsf{P}}, \boldsymbol{\delta}), \end{aligned}$$

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.

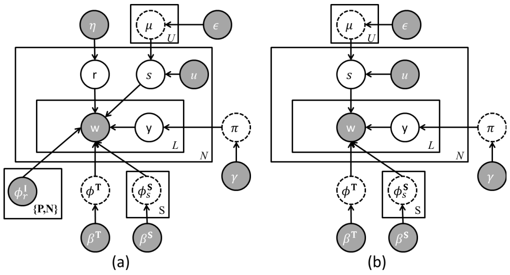

Figure 5: (a) DSM-1: A side clustering model that does not consider the interplay between interactions and sides. (b) DSM-2: A side clustering model that does not consider user interactions. Dashed variables will be collapsed out in Gibbs sampling.

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 [22]. With Gibbs sampling, we can deduce the following estimation:

$$\phi^{\mathsf{T}}_w = \frac{C^{\mathsf{T}}_w + \beta^{\mathsf{T}}}{\sum_{w=1}^{V} C^{\mathsf{T}}_w + V\beta^{\mathsf{T}}}. \quad \text{thread-word distr.} \quad (3)$$

$$\phi^{\mathsf{S}}_{s,w} = \frac{C^{\mathsf{S}}_{s,w} + \beta^{\mathsf{S}}}{\sum_{w=1}^{V} C^{\mathsf{S}}_{s,w} + V\beta^{\mathsf{S}}}. \quad \text{side-word distr.} \quad (4)$$

3.2 Models for Comparison

We study both degenerate models and existing approaches for comparison.

DSM-1: The model is in Figure 5(a). By comparing it to DSM, we evaluate the importance of adding the interplay between interactions and sides.

DSM-2: The model is in Figure 5(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 [20, 21] 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.

4. EXPERIMENTS

4.1 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 2.

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[5]. 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 [20]. 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.

	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 2: 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.

4.2 Interaction features

We first qualitatively analyze the interaction features discovered by our Interaction Model. We use the learned interaction word distribution in Eqn. (1). To visualize the interaction features, we adopt the approach used in [6]. The intuition is to downweight those features that are also popular under the other type of interactions.

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: 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. 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.[6] (2) N-grams tend to feature more identifiable expressions. E.g., `i agree` and `agree with you` show 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

[5] 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.

[6] 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.

negation features we can find more reasonable positive and negative interaction features to help infer interaction polarity.

4.3 Clustering

We evaluate our Debate Side Model on two debate side clustering tasks, i.e., post side clustering and user side clustering.

4.3.1 Clustering posts by sides

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 4. 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 4: 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.

4.3.2 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, 8, 12]. 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.

Results: We present the average performance of all the debates in Table 5. 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.

	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 5: 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.

4.4 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 6. 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.

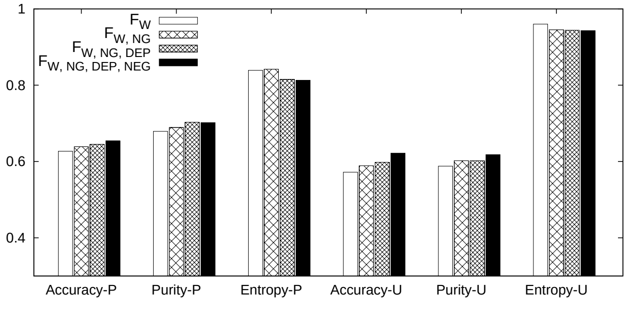

Figure 6: Impacts of different types of features on DSM in post side clustering ("-P") and user side clustering ("-U"). F_{W}, F_{NG}, F_{DEP}, and F_{NEG} stand for bag-of-words, N-gram, dependency relation and negation features respectively.

We have also studied adding polarity information to the opinionated features, the same as used in [21]. 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.

5. RELATED WORK

Finding interaction features is related to detecting agreement/ disagreement or contradiction from text. For this task, normally supervised methods are used [1, 10]. Besides, the argumentation theory has been used to recognize the entailment and contradiction relationships between two texts in [7]. In [4], the quotations are classified to specific topics and polarity (pro/con) using language models in debate corpus. A probabilistic model is studied in [19] to extract different types of expressions including agreement/disagreement expressions. In our work, 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 side clustering.

For the task of viewpoint finding, the work in [24] focused on identifying stances (sides) in online debates. They proposed a supervised approach for classifying stances in ideological debates relying on the discourse structure. An unsupervised method was studied in [23] which relies on associations of aspects with topics indicative of stances mined from the Web for the task. In comparison, our model is also an unsupervised one but we do not rely on any external knowledge except the interaction features mined from CreateDebate. The study in [20] proposed a probabilistic model to jointly model topics and viewpoints (sides). In their approach, they do not consider users. In comparison, our model particularly studies user interactions. A statistical model was presented in [16] for political discourse that incorporates both topics and viewpoints. Another work in [9] studied a model that also combines topics and viewpoints. These studies assume that documents are grouped by viewpoints, which is not the case for forum posts. Therefore, their models are not suitable for forum posts. A recent work [22] uses standard sentiment analysis to infer interaction polarities and models interplay between interactions and viewpoints. Differently, we infer the interaction polarity by using the interaction features learned by an interaction model which shows better performances than simple sentiment lexicon based method.

Another closely related task is subgroup detection, i.e. to cluster users holding similar viewpoints (sides). [2], [3], [8] and [12] study clustering-based approach for the task. Both textual content and social interactions are studied in [17] to find opposing network from online forums. In our experiments, we show that our model can also be used for subgroup detection, but meanwhile we also directly identify sides, which is not the goal of existing work on subgroup finding or opposing network extraction.

6. CONCLUSION

In this work, we study the task of clustering sides for posts or users for general threaded discussions. 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.

7. REFERENCES

[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 *Proc. of the Workshop on Language in Social Media (LSM 2011)*, pages 2–11, 2011.

[2] A. Abu-Jbara, P. Dasigi, M. Diab, and D. R. Radev. Subgroup detection in ideological discussions. In *Proc. of ACL*, pages 399–409, 2012.

[3] A. Abu-Jbara and D. R. Radev. Subgroup detector: A system for detecting subgroups in online discussions. In *ACL (System Demo)*, pages 133–138, 2012.

[4] R. Awadallah, M. Ramanath, and G. Weikum. Polaricq: polarity classification of political quotations. In *CIKM*, pages 1945–1949, New York, NY, USA, 2012. ACM.

[5] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*, 2010.

[6] D. M. Blei and J. D. Lafferty. *Topic Models*. Text Mining: Classification, Clustering, and Applications, Srivastava, Ashok N. and Sahami, Mehran, 2009.

[7] E. Cabrio and S. Villata. Combining textual entailment and argumentation theory for supporting online debates interactions. In *Proc. of the 50th ACL: Short Papers*, pages 208–212, Stroudsburg, PA, USA, 2012.

[8] 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 *Proc. of the 50th ACL*, pages 65–69, 2012.

[9] Y. Fang, L. Si, N. Somasundaram, and Z. Yu. Mining contrastive opinions on political texts using cross-perspective topic model. In *WSDM*, pages 63–72, 2012.

[10] 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 *Proc. of ACL'04*, pages 669–676, 2004.

[11] S. Greene and P. Resnik. More than words: syntactic packaging and implicit sentiment. In *Proc. of NAACL*, pages 503–511, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[12] A. Hassan, A. Abu-Jbara, and D. Radev. Detecting subgroups in online discussions by modeling positive and negative relations among participants. In *Proc. of EMNLP*, pages 59–70, 2012.

[13] M. Hu and B. Liu. Mining and summarizing customer reviews. In *Proc. of the 10th KDD*, pages 168–177, 2004.

[14] M. Joshi and C. Penstein-Rosé. Generalizing dependency features for opinion mining. In *Proc. of the ACL-IJCNLP 2009 Conference Short Papers*, pages 313–316, 2009.

[15] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *Proc. of the 41st ACL*, ACL '03, pages 423–430, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.

[16] W.-H. Lin, E. Xing, and A. Hauptmann. A joint topic and perspective model for ideological discourse. In *Proc. of ECML PKDD '08*, pages 17–32, Berlin, Heidelberg, 2008. Springer-Verlag.

[17] Y. Lu, H. Wang, C. Zhai, and D. Roth. Unsupervised discovery of opposing opinion networks from forum discussions. In *Proc. of the 21st CIKM*, pages 1642–1646, 2012.

[18] A. Mukherjee and B. Liu. Mining contentions from discussions and debates. In *Proc. of the 18th KDD*, pages 841–849, 2012.

[19] A. Mukherjee and B. Liu. Modeling review comments. In *Proc. of the 50th ACL*, pages 320–329, 2012.

[20] M. J. Paul and R. Girju. A two-dimensional topic-aspect model for discovering multi-faceted topics. In *AAAI*, 2010.

[21] M. J. Paul, C. Zhai, and R. Girju. Summarizing contrastive viewpoints in opinionated text. In *EMNLP*, pages 66–76, 2010.

[22] M. Qiu and J. Jiang. A latent variable model for viewpoint discovery from threaded forum posts. In *NAACL*, 2013.

[23] S. Somasundaran and J. Wiebe. Recognizing stances in online debates. In *Proc. of the ACL-IJCNLP*, pages 226–234, 2009.

[24] S. Somasundaran and J. Wiebe. Recognizing stances in ideological on-line debates. In *Proc. of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124, 2010.

[25] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proc. of the 2003 NAACL*, NAACL '03, pages 173–180, 2003.

[26] A. Tversky and D. Kahneman. The framing of decisions and the psychology of choice. *Science, New Series*, 211:453–458, 1981.

[27] T. Wilson, J. Wiebe, and P. Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *HLT/EMNLP*, 2005.