2014 IEEE International Conference on Data Mining

ShellMiner: Mining Organizational Phrases in Argumentative Texts in Social Media

Jianguang Du[†], Jing Jiang[‡], Liu Yang[§], Dandan Song[†,*], Lejian Liao[†]
† *School of Computer Science and Technology, Beijing Institute of Technology, Beijing, China*
‡ *School of Information Systems, Singapore Management University, Singapore*
§ *Center for Intelligent Information Retrieval, School of Computer Science, University of Massachusetts Amherst, USA*
Email: {dujianguang,sdd,liaolj}@bit.edu.cn; jingjiang@smu.edu.sg; lyang@cs.umass.edu

***Abstract*—Threaded debate forums have become one of the major social media platforms. Usually people argue with one another using not only claims and evidences about the topic under discussion but also language used to organize them, which we refer to as shell. In this paper, we study how to separate shell from topical contents using unsupervised methods. Along this line, we develop a latent variable model named Shell Topic Model (STM) to jointly model both topics and shell. Experiments on real online debate data show that our model can find both meaningful shell and topics. The results also show the effectiveness of our model by comparing it with several baselines in shell phrases extraction and document modeling.**

***Keywords*-argumentative text; organizational phrases; topic modeling; latent variable model**

I. Introduction

Argumentative texts are a type of text that aims to convince or persuade the audience. With the rapid growth of social media, we see more argumentative texts in discussion and debate forums contributed by ordinary users, where the topics under discussion range from religion to politics. An important aspect of mining social media content is to analyze and understand such argumentative texts.

In argumentative texts, when people present an argument, they often organize the evidences and claims in a certain way to strengthen the persuasion power. The organizational structures consist of phrases or sentence patterns that are not directly related to the issue under discussion but serve as connectors to link the evidences and claims. Consider the example below, which is a post from CreatDebate[1], a popular online debate forum.

> **I don't think that** this is necessarily accurate; **I do tend to agree that** science itself doesn't require absolute proof. **After all**, we don't have absolute proof of a number of theories. **Rather, I believe** it has more to do with the ability to make accurate predictions using the theory/hypothesis as a premise; hypotheses are yet untested, theories have passed numerous 'tests' and failed none that were not themselves fundamentally flawed.

As shown in the example, words in bold are organizational phrases that are not directly related to the topic under discussion but logically connects the text segments that are directly discussing the issue. Organizational phrases can be indicators of opinion expressions, such as *I don't think that*, or argumentative structures, such as *after all*. Generally, posts in discussion forums can be viewed as consisting of language used to express topical contents and language used to organize them.

Separating topical contents from organizational phrases can be beneficial for various tasks in mining argumentative texts from social media. For example, when we want to classify forum threads into general topics such as politics and religion, organizational phrases would not be very relevant and the classification presumably should be based mostly on the topical contents of the texts. On the other hand, identification of organizational phrases helps us understand the writer's writing styles and sometimes also the interactions between online users.

To the best of our knowledge, a recent study by Madnani et al. [1] is the first and only study that formally defines the task of identifying organizational text segments in argumentative texts. Following their terminology, we refer to organizational phrases as *shell* (and the rest of the texts as *meat*). In other words, shell is sequences of words used to organize topical contents.

However, the solutions provided in [1] have some limitations. One solution provided in [1] is a rule-based system using manually crafted rules. Since these rules are derived from a relatively small sample of texts, it is not clear how general they are when applied to different texts. Another solution provided in [1] is a supervised learning approach based on conditional random fields. This approach requires a sufficient amount of manually annotated data for training, which is expensive to obtain. Also, the training and evaluation in [1] is done on formal argumentative texts (student essays), but argumentative texts in online forums tend to be more informal and noisy.

In this paper, we propose a fully unsupervised model called Shell Topic Model (STM) to separate shell phrases from topical contents. In this model, a first-order (bigram) language model is used to model shell while topical contents are modeled using unigrams. In addition, we model function

[*] Corresponding author
[1] http://www.createdebate.com

1550-4786/14 $31.00 © 2014 IEEE
DOI 10.1109/ICDM.2014.98

IEEE computer society

words in topical contents. A switch variable is used to determine the type of a word, i.e. whether it is a shell word, a topical word or a function word.

To evaluate the usefulness of our model, we use the results of the model for the tasks of shell phrases extraction and document modeling. On a corpus of real discussion/debate forum posts from five domains on CreateDebate, STM can achieve encouraging improvement in all tasks compared with the baselines that do not separate shell from topical contents. The results confirm the usefulness of modeling shell inside argumentative texts, and also demonstrate the advantages of STM over existing topic models.

The contributions of this paper are summarized as follows:

1) We propose an unsupervised probabilistic model, named STM, to separate shell from topical contents.
2) With a real data set, we show that our model is able to identify both meaningful topics and shell.

II. Related Work

A. Topic models

Our work is about building language models from text. Since Blei et al. first introduced LDA [2], LDA has been extensively adopted and extended into more complex models for different purposes. Here we only cover some models that are closely related to our model.

Griffiths et al. [3] proposed an HMM-LDA model, which combines short-range syntactic dependencies along with long-range topical dependencies among the words in each document. In this model, it is assumed that words are generated from a number of hidden states that are related to each other through a first-order hidden Markov model. These states model the syntax of the language. One of the states is a special one that generates topic words. In effect, the hidden states model different parts of speech such as nouns, verbs, adjectives, adverbs and prepositions. The special topical state turns out to be mostly nouns. Our model bears strong resemblance to their model, but our focus is not on modeling the syntax of language. Instead, we focus on the task of modeling shell as well as hidden topics. In another model that combines HMM with LDA, Gruber et al. assumed that each sentence has only a single topic and the topic of the next sentence is more likely to be the same as the previous one [4]. This is a different HMM assumption than the one in [3] and in our work.

Bigram Topic Model [5] incorporated latent variables into bigram models based on hierarchical Dirichlet language model.This model can separate topic words from function words, such as "in," and "of," and make latent topics less dominated by function words compared with standard LDA. Although our model also identifies function words, we allow both bigrams and unigrams instead of only bigrams. Wang et al. [6] proposed a Structural Topic Model to find topical structures through modeling topical transitions using first order Markov chain. Du et al. [7] also captured topical structures, but the dependency between topics are modeled by Poisson Dirichlet Process (PDP). Wang et al. [8] proposed topical n-grams to discover topic phrases, such as "data mining" and "natural language processing." This model introduced a status variable to control whether to sample a unigram or bigram. Our model also has such a status variable, but we also capture function words, and the status variable is dependent on the previous status instead of previous word and topic.

B. Studies on online forums

There have been many studies on online discussion forums. A number of studies focus on mining opinions, viewpoints and stances from online forums [9; 10; 11]. Although some of them also use topic models, their objectives are very different from ours.

There have also been some work on finding phrases on debate/discussion forums. Mukherjee and Liu modeled agreement and disagreement indicator expressions to analyze user interactions [12]. Mukherjee et al. modeled tolerance in online discussions [13]. All these models use a variable trained with a Maximum Entropy model by utilizing part-of-speech (POS) information. In other words, their models are semi-supervised. In contract, our model is fully unsupervised. On the other hand, the phrases discovered in their models are mostly agreement and disagreement expressions, while our model can also find organizational phrases, such as "after all" in the example in Section I.

C. Argumentative analysis

Another line of studies related to our work is on argumentative analysis. Cabrio et al. [14] proposed a framework to support participants of online debates by combining textual entailment (TE) and argumentation theory. Guo et al. [15] proposed a weakly-supervised method to analyze argumentative structures of scientific papers. Rink et al. [16] proposed a generative model to discover semantic relations in electronic medical records. Xu et al. [17] proposed a framework to identify arguments from both intra-sentence and inter-sentence level. Feng et al. [18] defined several features to classify arguments by using argumentation schemes. One work in this line that shares a similar goal as ours is [1], where Conditional Random Field (CRF) is used to identify shell phrases based on several features. However, as we mentioned in Section I, their model is supervised and they focus on argumentative essays, which are more formal. In contract, our model are unsupervised, and we find shell on online debate forums, which likely contain more noisy words than student essays.

D. Discourse analysis

Part of our work is related to discourse analysis in natural language processing. Implicit discourse relation recognition [19; 20] tries to infer the implicit discourse relations given the raw text. For example, [20] identifies the implicit discourse relation of argument pairs. There are also other work that make use of discourse structures for different tasks, such as polarity analysis [21] and article wide temporal classification [22]. Another work [23] used Bayesian models to jointly model sentiment, aspects and discourse relations. While some shell phrases may be considered as indicators of discourse relations, such as "after all," our model can also discover other phrases, such as "I believe". On the other hand, our model simultaneously captures topics and transitions among status assigned to each word in a principled framework.

III. Shell Topic Model (STM)

A. Motivation

Assumptions about shell As mentioned in Section I, shell often consists of longer phrases than unigram words. In order to model shell, one possible way to do it is to view a document as bag of n-grams, where n can be any number. However, this greatly enlarges the vocabulary. Furthermore, many defined n-grams are meaningless. The other way is to label a set of training data to learn a model, such as Maximum Entropy (MaxEnt) model [12], to determine whether the current word can be an element of shell. However, training data are likely domain dependent, which makes them hard to re-use. Here we explore another approach. The idea is that we assume shell words are generated based on their previous words. To capture this assumption, we let each word have its own multinomial word distribution and a status variable (switch variable) to determine whether the current word is a shell word or non-shell word. So we can always find shell as bigrams in our model. Note that, longer shell phrases can be generated by concatenating consecutive bigrams.

Assumptions about status variables The other assumption we have is that shell words tend to be generated if the previous word is also a shell word, and function words tend to be generated after topical words. That is to say, the status variables of different words are dependent. To capture this assumption, we model transition probabilities between status variables. Specifically, in our model, a status variable is generated from a transition distribution associated with the previous status variable.

B. Model

Our model is an extension of LDA [2] but captures topical words, function words and shell words. To model them, we use a modified version of the HMM-LDA model [3]. In our model, we assume there are D documents in the corpus. For each document d $(1 \leq d \leq D)$ there are S_d sentences. For each sentence s $(1 \leq d \leq S_d)$ there are $N_{d,s}$ words. Let $w_{d,s,n}$, an index between 1 and V, denote the n^{th} word in sentence s in document d, where V is the vocabulary size. Each word is associated with a status variable $x \in \{0, 1, 2\}$, which represents shell status, topical status, and function status, respectively. Each status generates words according to a multinomial word distribution. Both topical status and function status are used to generate unigram words. While topical status is used to model words that provide semantic content, function status is used to model words that serve syntactic functions. In contrast, shell status is used to model shell which consists of organizational phrases, such as "I do believe that." We should notice that shell status can model bigrams and successive bigrams can form longer shell phrases. Because those function words and shell words are not topic specific, we model them globally. Like in standard LDA, we assume that each topic k has a mutlinomial word distribution ϕ_k^t. In addition, we assume the function status is associated with a global multinomial word distribution ϕ^s, and each shell status is associated with a specific global mutlinomial word distribution ψ_v for each word v. We assume that ϕ_k^t and ϕ^s are sampled from a Dirichlet prior distribution with parameter β, and ψ_v are sampled from another Dirchlet distribution with parameter γ.

We then model distributions for each status. We assume that there are transition probabilities between statuses, and the transition probabilities from a status to other statuses follow a multinomial distribution. We use σ_x to denote the initial and transition probabilities for status x. We also assume that these distributions are sampled from a Dirichlet distribution with parameter δ.

Finally, we model the generation process of a document. Like in standard LDA, we assume a topic distribution θ^d for each document d, which is also sampled from a Dirichlet distribution with parameter α. For each word $w_{d,s,n}$ in sentence s in document d, we first sample its status variable $x_{d,s,n}$ from $\sigma_{x_{d,s,n-1}}$ and its topic $z_{d,s,n}$ from θ^d. We then sample a word according to the corresponding word distribution.

The document generative process can be described as follows:

```
• Draw $\phi^s \sim Dir(\beta)$
• For each topic $k = 1, ..., T$, draw $\phi_k^t \sim Dir(\beta)$
• For each status $x = 0, 1, 2$, draw $\sigma_x \sim Dir(\delta)$
• For each word $w = 1, ..., V$, draw $\psi_w \sim Dir(\gamma)$
• For each document $d = 1, ..., D$
    – Draw $\theta^d \sim Dir(\alpha)$
    – For each word $n = 1, ..., N_{d,s}$ in each sentence $s = 1, ..., S_d$
        ◇ Draw $x_{d,s,n} \sim Multi(\sigma_{x_{d,s,n-1}})$
        ◇ Draw $z_{d,s,n} \sim Multi(\theta^d)$
        ◇ if $(x_{d,s,n} = 0)$            // shell status
            - Draw $w_{d,s,n} \sim Multi(\psi_{w_{d,s,n-1}})$
        ◇ else if $(x_{d,s,n} = 1)$       // topical status
            - Draw $w_{d,s,n} \sim Multi(\phi_{z_{d,s,n}}^t)$
        ◇ else                          // function status
            - Draw $w_{d,s,n} \sim Multi(\phi^s)$
```

Note that at the beginning of a sentence we only allow function status and topical status. That is to say, only unigram is allowed at the beginning of a sentence. The graphical presentation for our model is shown in Fig. 1.

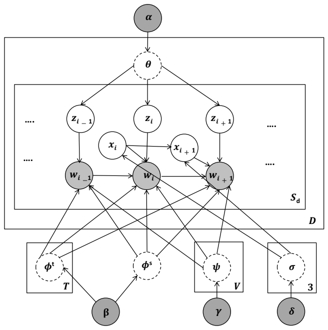


Figure 1. The plate notation of STM. Dashed variables will be collapsed out in Gibbs sampling.

C. Inference

Exact inference of hierarchical Bayesian models is intractable due to large number of variables and parameters. Approximate inference methods have been developed to solve this problem, such as Gibbs sampling [24], expectation propagation and Gibbs-EM. In this paper we use collapsed Gibbs sampling [24] to estimate the model parameters. The parameters θ, ϕ^t, ϕ^s, ψ and σ can be integrated out by using collapsed Gibbs sampling. To perform Gibbs sampling, we need to work out the conditional probabilities $p(x_{d,s,n}, z_{d,s,n}|\mathbf{w}, \mathbf{x}_{-(d,s,n)}, \mathbf{z}_{-(d,s,n)}, \alpha, \beta, \gamma, \delta)$, where $\mathbf{x}_{-(d,s,n)}$ represents the set of all $\mathbf{x}$ except $x_{d,s,n}$ and $\mathbf{z}_{-(d,s,n)}$ is defined similarly. In each step of Gibbs sampling, we jointly sample status $x_{d,s,n}$ and topic $z_{d,s,n}$ based on the current assignments of all the other hidden variables and all the observed words. Due to space limit, we leave out the details of the Gibbs sampling formulas.

IV. Experiment

For evaluation, we conducted both quantitative and qualitative experiments as we will present in this section.

A. Data Set and Experiment Settings

Data: We use a debate forum data set crawled from CreateDebate. The data set contains threads in different domains on different topics such as "Does God exist?" Each thread contains many posts written by different users expressing their opinions and commenting on each other's posts. We discard threads whose size is smaller than 5K, since these threads are not very popular and thus contain less information. We also discard those domains having less than 40 threads. Finally, we get 775 threads from 5 domains. We also split each thread into sentences because we model word dependencies at the sentence level. Finally, our data set consists of 44,667 unique words in the vocabulary, 3,389,240 words, and 201,452 sentences with an average of 16.8 words per sentence[2]. Note that we treat each thread as a document.

Table III
Precision @ top 10, 20, 30, 40, 50 and 60 rank positions of STM and baseline.

n-grams	Methods	p@10	p@20	p@30	p@40	p@50	p@60
4-grams	STM	**1.000**	**0.900**	**0.833**	**0.875**	**0.860**	**0.833**
	Baseline	0.600	0.800	0.767	0.800	0.780	0.800
5-grams	STM	**0.900**	**0.850**	**0.800**	**0.825**	**0.780**	**0.750**
	Baseline	0.700	0.600	0.600	0.625	0.640	0.667

Parameter settings: In all our experiments, we fix the number of topics $T = 50$, and empirically set the Dirichlet priors as follows: $\alpha = 50.0/T$, $\beta = 0.1$, $\delta = 0.5$, and $\gamma = 0.1$ as suggested in [24]. We run STM with 500 iterations of Gibbs sampling.

B. Qualitative Evaluation

As mentioned in Section III, our model can identify both topical words and shell phrases. In this section, we will show the hidden topics and shell phrases learned by our model. Note that although our model can only discover bigrams as shell, we can concentrate continues bigrams to form higher order n-grams shell.

We first show the topics discovered by our model. We randomly choose 10 topics and observe the top 10 words learned by our proposed STM model. The results are shown in Table I. We can see from the table that the discovered topics generally make sense. For example, from the top words like "music," "rap" and "songs," it is easy to justify that Topic 9 is about music. Top topical words like "religion," "islam," "christianity" and "muslims" demonstrate Topic 5 is about religion. We also observe a few noisy words in some topics, such as "was" in Topic 1 and "him" in Topic 7. This is because we do not remove stop words in our model, and they are general words that have high frequency. However, as topical words mentioned above, although there is a few noisy words, we can still identify each topic.

We then show the shell phrases learned by our model. For comparison, we also show the most frequent 4-grams and 5-grams in our data set. We denote this method as TFP (top frequent phrases). The results are shown in Table II. We can see that most of the top 4-grams and 5-grams discovered by STM are indeed shell, such as "i do n't believe," "i 'm pretty sure" and "i do n't agree with." On the other hand, those n-grams discovered by TFP contain more noisy phrases, such as "can yes we can," "there is no god" and "do n't believe in god."

C. Quantitative Evaluation

1) Shell Phrases Extraction: In order to quantitatively evaluate the shell phrases we can identify using our model, we use human annotated ground truth to measure the performance. Using both STM and a baseline which simply ranks 4-grams and 5-grams by frequency, we first obtain two lists of the top shell phrases by each method, respectively. Here we choose the top 100 4-grams and 5-grams. We then take

[2] Code and data are available at: https://www.dropbox.com/s/ph7666nfuykgddq/ShellMiner-data-code.zip?dl=0

Table I
TOP WORDS FOR DIFFERENT TOPICS DISCOVERED BY STM.

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5	Topic 6	Topic 7	Topic 8	Topic 9	Topic 10
war	stalin	gay	hitler	religion	israel	god	life	music	america
military	russia	marriage	jews	islam	the	him	space	rap	britain
iraq	during	women	germany	christianity	hamas	evil	earth	metal	liberals
was	million	homosexuality	german	muslims	palestine	us	end	like	country
wars	production	homosexuals	propaganda	christians	palestinians	good	planet	dubstep	conservatives
country	famine	men	liberalism	morality	land	all	planets	songs	history
the	success	sex	fascism	religious	jews	jesus	aliens	song	patriotism
us	deaths	gays	communists	religions	palestinian	love	infinite	rock	germany
soldiers	grain	children	crazy	christian	gaza	christianity	travel	listen	nation
terrorists	collectivization	homosexual	himself	beliefs	state	man	universe	band	liberal

Table II
TOP 4-GRAMS AND 5-GRAMS DISCOVERED BY STM AND TFP (TOP FREQUENT PHRASES).

4-grams		5-grams	
STM	TFP	STM	TFP
i do n't think	i do n't think	i do n't see how	we can yes we can
i do n't know	i do n't know	i do n't believe in	yes we can yes we
in the first place	can yes we can	i do n't think that	can yes we can yes
i do n't believe	yes we can yes	i do n't agree with	has nothing to do with
i do n't see	we can yes we	i do n't know why	that there is no god
i 'm not sure	if you do n't	there is no such thing	there is no such thing
i do n't care	i do n't believe	you do n't know what	i do n't believe in
i do n't have	there is no god	i do n't understand why	i do n't think that
you do n't have	in the first place	most important entitles individual citizens	you do n't believe in
there is no evidence	do n't believe in	i have no problem with	i do n't see how

the union of these two lists of phrases and present them to two human judges. The two human judges independently label these phrases as either shell or non-shell. The criterion we give to the human judges is that if a phrase can be used in an argument regardless of the debate topic, then the phrase is considered shell. The agreement score between the two judges using Cohen's kappa is 0.631, indicating substantial agreement. We then discard those phrases which have conflicting labels. The remaining phrases are used to compute precision @ n ($p@n$), a commonly used metric for ranking problems such as document retrieval. We set n to 10, 20, 30, 40, 50 and 60. The results are shown in Table III. We can see that for both 4-grams and 5-grams, STM consistently outperforms the baseline for all n.

2) Document Modeling: To measure whether STM is a good generative model, we also compute the perplexity on unseen test set. We compare with the perplexity values of standard LDA.

Perplexity, a commonly used metric to asses the predictive power of a new model, is algebraically equivalent to the inverse of the geometric mean per-word likelihood [2]. Formally, for a test set of M documents, the perplexity is defined as follows:

$$\textit{perplexity}(D_{\textit{test}}) = \exp\Big(- \frac{\sum_{d=1}^{M} \log p(\mathbf{w}_d)}{\sum_{d=1}^{M} N_d} \Big). \tag{1}$$

Here, $\mathbf{w}_d$ represents all the words in document d, and N_d is the number of words in document d. The definition of perplexity implies that the lower the perplexity is, the better the model. Because higher per-word likelihood on the test set means that the model can predict the unseen data better. According to Eqn 1, given the learned parameters $\hat{\theta}^d$ (document-topic distributions) and $\hat{\varphi}_k$ (topic-word distributions), the perplexity of LDA is calculated by:

$$\begin{aligned} & \textit{perplexity}(D_{\textit{test}}) \\ = & \exp\Big(- \frac{\sum_{d=1}^{M} \sum_{n=1}^{N_d} \log(\sum_{k=1}^{T} \hat{\varphi}_{k,w_{d,n}} \hat{\theta}^d_k)}{\sum_{d=1}^{M} N_d} \Big) \end{aligned} \tag{2}$$

Similarly, the perplexity of STM is given by:

$$\begin{aligned} & \textit{perplexity}(D_{\textit{test}}) \\ = & \exp\Big(- \frac{\sum_{d=1}^{M} \sum_{s=1}^{S_d} \sum_{n=1}^{N_{d,s}} \log p(w_{d,s,n})}{\sum_{d=1}^{M} N_d} \Big) \end{aligned} \tag{3}$$

where

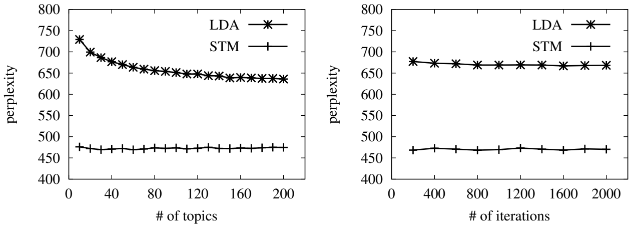
$$\begin{aligned} & p(w_{d,s,n}|n \neq 1) = \sum_{s=0}^{2} p(x_{d,s,n-1} = s) \times \\ & (\hat{\psi}_{w_{d,s,n-1},w_{d,s,n}} \hat{\sigma}_{s,0} + \hat{\theta}^d_k \hat{\phi^t}_{k,w_{d,s,n}} \hat{\sigma}_{s,1} + \hat{\phi^s}_{w_{d,s,n}} \hat{\sigma}_{s,2}) \end{aligned} \tag{4}$$

and

$$p(w_{d,s,1}) = \hat{\theta}^d_k \hat{\phi^t}_{k,w_{d,s,1}} \hat{p}(x=1) + \hat{\phi^s}_{w_{d,s,1}} \hat{p}(x=2). \tag{5}$$

Here, $\hat{p}(x = s)$ means the estimated probability that the status of the current word is s, and it can be estimated by the ratio of status s appeared at the first word of all sentences in the training set. Note that at the beginning of a sentence, the status variable x is forced to be either 1 (topical status) or 2 (function status).

In our experiment, we used 20% of our data set as test set, and the rest 80% data for training the model. For LDA, we empirically set Dirichlet parameters as $\beta = 0.01$, $\alpha =$



(a) (b)

Figure 2. Perplexity results for STM and LDA with different number of (a) topics and (b) iterations.

$50.0/T$, and train the model using the popular open source topic modeling package JGibbLDA[3].

Increasing the number of topics

First, we compare perplexity values of LDA and our STM model across the number of topics with 500 iterations. The results are shown in Fig. 2(a). We can see that as the number of topics increases, the perplexity of LDA drops. However, the perplexity of STM does not change significantly, which means the predictive power of STM is stable when the number of topics changes. We can also see that our STM model achieves significantly lower perplexities with different topics showing that STM fits the debate/discussion forum data better.

Increasing the number of iterations

We then fix the number of topics to 50, and increase the number of iterations. We can see from Fig. 2(b) that as the number of iterations increases, the perplexity results of both LDA and STM do not change much. One possible reason is that our data set is not very large, which leads both models to convergence with a few iterations. Also, STM outperforms LDA under all different settings of iterations.

V. Conclusion

In this paper, we proposed a novel latent variable model called Shell Topic Model (STM) to mine debate/discussion forums data. Our model is based on the observation that users tend to express their opinions using not only evidence and claims (topics), but also organizational phrases (shell) to organize them. Our model captures the observations in an unsupervised way to jointly model shell, topics, and function words. In particular, to capture shell which are usually long-term phrases, we model shell as bigrams and concatenate consecutive bigrams to form longer phrases. Experimental results showed that our model outperformed the baseline in the task of shell phrases extraction. Statistical experiments of perplexity are also conducted showing that our proposed STM model fit the data better and find both meaningful topics and shell.

In this work, we only considered content of forum data. As some previous work has shown, user attributes [10] may improve the performance of topic models. We hope to explore this direction in our future work. Another interesting aspect is the interaction features, such as replying and quoting, on forums. We also plan to incorporate these features in the future work.

Acknowledgment

This work was done during Jianguang Du's visit to Singapore Management University. This work was partially funded by the National Program on Key Basic Research Project (973 Program, Grant No. 2013CB329605), National Natural Science Foundation of China (NSFC, Grant Nos. 61472040 and 60873237), and Beijing Higher Education Young Elite Teacher Project (Grant No. YETP1198).

References

[1] N. Madnani, M. Heilman, J. Tetreault, and M. Chodorow, "Identifying high-level organizational elements in argumentative discourse," in *NAACL-HLT*, 2012, pp. 20–28.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *JMLR*, vol. 3, pp. 993–1022, 2003.

[3] T. L. Griffiths, M. Steyvers, D. M. Blei, and J. B. Tenenbaum, "Integrating topics and syntax." in *NIPS*, 2004, pp. 537–544.

[4] A. Gruber, Y. Weiss, and M. Rosen-Zvi, "Hidden topic markov models," in *AISTATS*, 2007, pp. 163–170.

[5] H. M. Wallach, "Topic modeling: beyond bag-of-words," in *ICML*, 2006, pp. 977–984.

[6] H. Wang, D. Zhang, and C. Zhai, "Structural topic model for latent topical structure analysis," in *ACL*, 2011, pp. 1526–1535.

[7] L. Du, W. L. Buntine, and H. Jin, "Sequential latent dirichlet allocation: Discover underlying topic structures within a document," in *ICDM*, 2010, pp. 148–157.

[8] X. Wang, A. McCallum, and X. Wei, "Topical n-grams: Phrase and topic discovery, with an application to information retrieval," in *ICDM*, 2007, pp. 697–702.

[9] A. Abu-Jbara, M. Diab, P. Dasigi, and D. Radev, "Subgroup detection in ideological discussions," in *ACL*, 2012, pp. 399–409.

[10] M. Qiu, L. Yang, and J. Jiang, "Modeling interaction features for debate side clustering," in *CIKM*, 2013, pp. 873–878.

[11] S. Gottipati, M. Qiu, Y. Sim, J. Jiang, and N. A. Smith, "Learning topics and positions from debatepedia," in *EMNLP*, 2013.

[12] A. Mukherjee and B. Liu, "Mining contentions from discussions and debates," in *KDD*, 2012, pp. 841–849.

[13] A. Mukherjee, V. Venkataraman, B. Liu, and S. Meraz, "Public dialogue: Analysis of tolerance in online discussions," in *ACL*, 2013.

[14] E. Cabrio and S. Villata, "Combining textual entailment and argumentation theory for supporting online debates interactions," in *ACL*, 2012, pp. 208–212.

[15] Y. Guo, A. Korhonen, and T. Poibeau, "A weakly-supervised approach to argumentative zoning of scientific documents," in *EMNLP*, 2011, pp. 273–283.

[16] B. Rink and S. Harabagiu, "A generative model for unsupervised discovery of relations and argument classes from clinical texts," in *EMNLP*, 2011, pp. 519–528.

[17] X. Fan, Z. Qiaoming, and Z. Guodong, "A unified framework for discourse argument identification via shallow semantic parsing," in *COLING*, 2012, pp. 1331–1340.

[18] V. W. Feng and G. Hirst, "Classifying arguments by scheme," in *ACL*, 2011, pp. 987–996.

[19] E. Pitler, A. Louis, and A. Nenkova, "Automatic sense prediction for implicit discourse relations in text," in *ACL*, 2009, pp. 683–691.

[20] Y. Hong, X. Zhou, T. Che, J. Yao, Q. Zhu, and G. Zhou, "Cross-argument inference for implicit discourse relation recognition," in *CIKM*, 2012, pp. 295–304.

[21] B. Heerschop, F. Goossen, A. Hogenboom, F. Frasincar, U. Kaymak, and F. de Jong, "Polarity analysis of texts using discourse structure," in *CIKM*, 2011, pp. 1061–1070.

[22] J.-P. Ng, M.-Y. Kan, Z. Lin, W. Feng, B. Chen, J. Su, and C.-L. Tan, "Exploiting discourse analysis for article-wide temporal classification," in *EMNLP*, 2013, pp. 12–23.

[23] A. Lazaridou, I. Titov, and C. Sporleder, "A bayesian model for joint unsupervised induction of sentiment, aspect and discourse representations," in *ACL*, 2013.

[24] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National academy of Sciences of the United States of America*, vol. 101, no. Suppl 1, pp. 5228–5235, 2004.

[3] http://jgibblda.sourceforge.net/