Intensity of Relationship Between Words: Using Word Triangles in Topic Discovery for Short Texts

Ming Xu, Yang Cai, Hesheng Wu, Chongjun Wang(✉), and Ning Li

National Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
xuming0830@gmail.com, caiyang023@gmail.com, weierson@smail.nju.edu.cn, {chjwang,ln}@nju.edu.cn

Abstract. Uncovering latent topics from given texts is an important task to help people understand excess heavy information. This has caused the hot study on topic model. However, the main texts available daily are short, thus traditional topic models may not perform well because of data sparsity. Popular models for short texts concentrate on word co-occurrence patterns in the corpus. However, they do not consider the intensity of relationship between words. So we propose the new way, called word-network triangle topic model (WTTM). In WTTM, we search for the word triangles to measure the relations between words. The results of experiments on real-world corpus show that our method performs better in several evaluation ways.

Keywords: Topic model · Short texts · Word co-occurrence network · Document classification

1 Introduction

Recent years, the technology of Internet is growing rapidly, people are more willing to get information from the Internet. Information surplus becomes a big problem. So topic distillation is crucial for many content analysis tasks. And a popular research field is topic model.

Topic model [4] is used to discover latent topics from given texts. It assumes that a document is generated from a mixture of topics, where a topic is a probabilistic distribution over words. Topic model is helpful in automatically extracting thematic information from large archive [1]. It's widely used in modeling text collections like news articles, research papers and blogs. Now the popular models are PLSA (also as PLSI) [5] and LDA [2].

However, with the great popularity of social networks and Q&A networks, short texts occupy the main space on the Web. Inferring latent topics based on these texts may help a lot in daily life. However, a bad news is that traditional topic models do not perform well in these situations. The main reason may

L. Chen et al. (Eds.): APWeb-WAIM 2017, Part I, LNCS 10366, pp. 642–649, 2017.
DOI: 10.1007/978-3-319-63579-8_48

rely on that the parameters of topic model are calculated according to the co-occurrence of words in the texts, while short texts is short of such relations because of lack of words [6].

Under the circumstances, we need some improvements on topic model for short texts. One simple but useful idea is to bring together the similar short documents to structure longer documents. For example, combining the texts written by the same user in twitter [9] or combining the texts which own the same words [6]. Some others add extra information into short texts to enrich data density. However, methods above are highly data-dependent or task-dependent. Another popular approach now is to extend the documents based on the information from original texts, like BTM [10], RIBS-TM [7] and WNTM [11]. Though these models have good performance on the short texts, they ignore that the relations of the word co-occurrence patterns are not all so close. As we can see, not any two words in the documents share similarly close relation.

To solve the data sparsity on short texts and value relationship between words, we propose the method called $Word-network\ Triangle\ Topic\ Model$ (WTTM). The method came from the thoughts below. (1) Though topic model infers the parameters based on the word co-occurrence patterns, but not all patterns share the same intensity of relations. (2) How can we express the closer relationship among words as the word pairs do not show the intensity of relationship. Triangle is an important structure in graph theory. It represents strong relationship among the nodes in the graph. So we choose triangles as our basic structures. In WTTM, we build up the global word co-occurrence network of word pairs in the corpus, and find out the word triangles in the network. Then we make use of word triangles we find to train the models. Compared with the existing models, we use the word triangles to draw the close relationship among words. In this way, we reduce the influence of the weak-related patterns and strengthen the highly-related patterns.

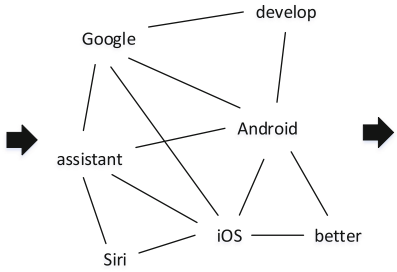
Extensive experiments are conducted on real-world dataset to compare WTTM with the baseline models. Through the experiments on topic coherence and document semantic classification, the outstanding results prove that our method is a good choice of topic inferring for short texts.

The rest of the paper is organized as follows. We first give the detail analysis of our approach is followed in Sect. 2; Experimental results are explained in Sect. 3. At last, we'll conclude our present work in Sect. 4 and some measures to perfect the method are also pointed out.

2 Our Approach

Probabilistic topic models find the latent topics of the given documents based on word co-occurrence patterns in the documents, thus the results will be highly effected by data sparsity in short texts scenario. To overcome the difficulty, we came up with the idea called word-network triangle topic model (WTTM), which chooses the more related patterns and represent them with word triangles according to the word co-occurrence network from the corpus. The details will be described as follows.

index	doc
1	Google assistant for Android
2	Siri is the assistant for iOS
3	Google assistant is only for Android instead of iOS
4	Android is developed by Google
5	Android is better than iOS



biterm	indexSet
Google,develop	4
Google,assistant	1,3
Google,Android	1,3,4
Google,iOS	3
assistant,Siri	2
assistant,iOS	2
assistant,Android	3
Siri,iOS	2
iOS,Android	3,5
iOS,better	5
Android,better	5
Android,develop	4

Fig. 1. A example of the word co-occurrence network from the input corpus

2.1 Word Co-occurrence Network

The word co-occurrence network (which is denoted as word network in the following text) is a global undirected graph of given texts. In the word network, nodes are the words appear in the corpus and an edge between two nodes shows that the two words appear in the same document. It's known that topic model is based on the bag of words model so it's possible for us to build up the global network of words in the corpus. Figure 1 shows an example of word network.

Definition 1. *Given a set of documents* $D = \{d_1, d_2, \ldots, d_n\}$*, and every document* d_k *is divided a group of words,* $d_k = \{w_1, w_2, \ldots, w_m\}$*. The word co-occurrence network is represented as* $N = <V, E, S>$*.*
where:
V *represents the nodes in the network* N*, namely the whole group of words in the given documents* D*.* $V = \{w_i | w_i \in d_k\}$*.*
E *represents the edges in the network* N*, namely the word co-occurrence patterns in the given documents* D*.* $E = \{e_{ij} = (w_i, w_j) | w_i, w_j \in d_k\}$*.*
$S_{e_{ij}}$ *represents the set of indexes of the edge* e_{ij}*. It expresses indexes of the documents where the words* w_i, w_j *can be found in* D*.* $S_{e_{ij}} = \{k | e_{ij} = (w_i, w_j) \wedge w_i, w_j \in d_k\}$*.*
S *is the group of all the index sets of every edge* $e_{ij} \in E$*.* $S = \{S_{e_{ij}} | e_{ij} \in E\}$*.*

In our idea, the co-occurrence of two words in the same context shows that the two words share a relationship. And we tag the edges with the indexes of the documents where the connected words co-occurred. It's obvious that the more tags of indexes one edge owns, the more triangles it may be chosen in. Based on the co-occurrence patterns, we build up the word network.

2.2 Word Triangle

With the help of the word network we built up above, we can find the triangle connections among the words in the corpus.

Traditional topic models take advantage of the word co-occurrence patterns by modeling word generation from the document level. However, patterns do not show the relations, but the intensity of relations of patterns should be taken into consideration and we need to choose a new structure to perform strong relations. Triangle is an important structure in graph theory. A triangle is a set of three nodes that are pairwise connected and is arguably one of the most important patterns in terms of understanding the inter-connectivity of nodes in real graphs [3]. And a triangle is the complete graph, that's to say nodes in a triangle share closer relationships than common graphs. So here we came up with the idea of valuing the relation of word co-occurrence patterns and using word triangles to express the closer relations.

Definition 2. *Given the word co-occurrence network* $N = <V, E, S>$*, for every three words* $w_i, w_j, w_k \in V$*, if:*

(1) $e_{ij} = (w_i, w_j), e_{ik} = (w_i, w_k), e_{jk} = (w_j, w_k) \wedge e_{ij}, e_{ik}, e_{jk} \in E$.
(2) $s_{e_{ij}}, s_{e_{ik}}, s_{e_{jk}} \in S$, $s_{e_{ij}} \neq s_{e_{ik}} \wedge s_{e_{ij}} \neq s_{e_{jk}} \wedge s_{e_{ik}} \neq s_{e_{jk}}$.

the three words w_i, w_j, w_k *can structure the word triangle* $t = (w_i, w_j, w_k)$.

We emphasize that the edges are labeled with different sets of tags to avoid the situation that any group of three words in the same context will be regarded as a word triangle. In our experiments we find that there are few words which are not included in any triangle, we tend to think they are unrelated in the corpus and ignore them.

2.3 Word-Network Triangle Topic Model

The key idea of BTM is to learn the latent topics over given short texts based on the aggregated word patterns in the whole corpus. However, relation of some patterns may not be so close. So we use triangles to evaluate patterns and exclude weak-related ones. In our model, each word triangle is drawn from a specific topic independently. And the probability of the triangle drawn from a specific topic is determined by the proportion that the three words in the triangle to be drawn from the topic. Suppose α and β are the Dirichlet priors. And the process of WTTM can be described as follows:

1. For each topic z
   (a) draw a topic-specific word distribution $\phi_z \sim Dir(\beta)$
2. Draw a topic distribution $\theta \sim Dir(\alpha)$ for the whole collection
3. For each triangle t in the triangle set T
   (a) draw a topic assignment $z \sim Multi(\theta)$
   (b) draw three words: $w_i, w_j, w_k \sim Multi(\phi_z)$

According to the procedure above, the probability of a word triangle $t = (w_i, w_j, w_k)$ can be calculated as:

$$P(t) = \sum_z P(z)P(w_i|z)P(w_j|z)P(w_k|z) = \sum_z \theta_z \phi_{i|z} \phi_{j|z} \phi_{k|z} \tag{1}$$

And the possibility of the given texts is:

$$P(T) = \prod_{i,j,k} \sum_{z} \theta_z \phi_{i|z} \phi_{j|z} \phi_{k|z} \tag{2}$$

From the process we directly train the model using the word triangles as basic elements of the topics. We consider the close relationship of triangles to select the stronger relationships among words. What's more, the whole patterns in the corpus are aggregated to find the triangle connections. In consequence, we evaluate the patterns in the corpus to reveal the topics better. To train the parameters mentioned above, we adopt Gibbs sampling the same as BTM.

2.4 Inferring Topics for a Document

From the previous introduction of the procedure we can know that WTTM trains the model based on the whole corpus. Therefore, the training results do not give the topic distribution on the documents directly. We need some efforts to obtain the topics of every document and the distribution. The calculation is as follows:

We assume that the topic distribution of the document is the same as the expectation of the topic distribution of the group of word triangles containing the word patterns in the document:

$$P(z|d) = \sum_{t} P(z|t) P(t|d) \tag{3}$$

In the formula above, $P(z|t)$ can be calculated according to Bayes' formula based on the distribution of words in topics $P(w|z)$ and the distribution of topics in the corpus $P(z)$ got from WTTM:

$$P(z|t) = \frac{P(z)P(w_i|z)P(w_j|z)P(w_k|z)}{\sum_z P(z)P(w_i|z)P(w_j|z)P(w_k|z)} = \frac{\theta_z \phi_{i|z} \phi_{j|z} \phi_{k|z}}{\sum_z \theta_z \phi_{i|z} \phi_{j|z} \phi_{k|z}} \tag{4}$$

where $P(z) = \theta_z$ and $P(w_i|z) = \phi_{i|z}$.

And the other one to calculate is $P(t|d)$. We can get it easily based on the distribution of word triangles in the document:

$$P(t|d) = \frac{n_d(t)}{\sum_t n_d(t)} \tag{5}$$

where $n_d(t)$ is the frequency of the word triangle t of the group of word triangles in the document d.

And for the documents which structure no word triangle, we infer the topic distribution of such document using the expectation of the topic distributions of words in the documents. According to the formulas above, we can get the topic distribution for given documents.

3 Experiments

In this section, we conduct experiments in the real-world short texts collection of questions with labels from Zhihu to verify the improvements on the algorithm. We take two typical topic models, BTM and LDA, to perform our comparison. Besides, as BTM and WTTM do not model the generation process of documents so we cannot evaluate the models by calculating the perplexity. In consequence, we evaluate the performance of WTTM from two sides, topic coherence and semantic document classification.

3.1 Topic Coherence

In order to compare the methods with quantitative metrics, we introduce the topic coherence to measure the relations of words in the topic we get.

Topic coherence, also called UMass measure, is proposed by Mimno et al. [8] in 2011 for topic quality evaluation. Topic coherence weighs the semantic similarity degree of the words with high-frequency in the topics learned from the documents and then decides the quality of the model. The higher the topic coherence is, the better the topic quality is.

We compare WTTM with LDA and BTM on the short text collection. For all the models, we change the number of topics from 10 to 50. And the average results of topic coherence of the three models are listed in Table 1. Here the number of topics is 30 (P-value < 0.001 by T-test):

Table 1. Average results of topic coherence on the top K words in topics inferred by LDA, BTM, and WTTM on Zhihu questions.

T	5	10	20
LDA	-60.1 ± 4.5	-442.3 ± 18.9	-2752.2 ± 62.4
BTM	-58.0 ± 6.0	-365.7 ± 16.2	-2112.7 ± 30.1
WTTM	-37.4 ± 0.9	-200.0 ± 3.8	-1215.2 ± 17.7

T in Table 1 refers to the number of top words we choose in the topic, ranging from 5 to 20. From the table we find that the average topic coherence of WTTM is obviously better than BTM and LDA. In fact, LDA infers latent topics based on the limited word co-occurrence patterns in each document. So LDA cannot learn the topics of short texts accurately as lack of words. BTM outperforms LDA as it learns topics by modeling directly on the word co-occurrence patterns and overcomes the data sparsity. However, BTM doesn't evaluate the patterns and some weak-related patterns may result in bad influence. Our method exclude such patterns and uses word triangles to express the strong relations. That's the reason why WTTM performs better than BTM. Comparing with the two baseline methods, WTTM can find more related words in one topic.

3.2 Semantic Document Classification

To compare the topic distributions in documents which can express the short documents more accurately, we conduct the experiments to train the document classifiers on the question collection of Zhihu in this part. The topic distributions in each document can be used for dimensionality reduction, that's to say the documents can be represented with a set of topics which can express the contents well. Then we can use the topics and the proportion of topics as the feature vectors to express the documents. So it's convenient for us to value the accuracy of the topical representation of short documents for classification. In the experiments, splitspilt into training subset and testing subset with the ratio 9:1, then we classified them by using random forest in Weka. The number of trees is 50 and the max depth of each tree is 20. We show the accuracy on 10-fold cross validation in Fig. 2.

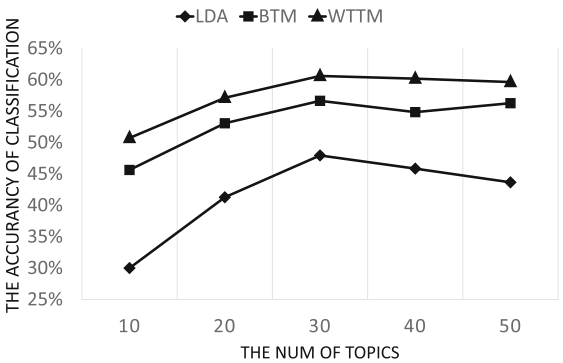


Fig. 2. The accuracy of document classification of WTTM, BTM, and LDA on Zhihu questions

From the results of Fig. 2, we find that WTTM outperforms BTM and LDA in classifying short texts. The results shows that the topics which WTTM infers can represent the documents better. We can figure out again that LDA is not good at topic inferring of short documents. And the reason is similar as what is said in experiments on topic coherence. The data sparsity in short texts affects the results of LDA. It shows that the topics which WTTM infers can represent the documents better. BTM performs better than LDA. WTTM performs better than BTM as it emphasizes the evaluation of the relations between words in the corpus. So we can conculde that WTTM can find better topics to express the texts.

4 Conclusion

Topic inference on short texts is becoming increasingly popular. In this paper, we proposed a new topic model for short texts, namely word-network triangle topic

model (WTTM), which finds the word triangles in the word co-occurrence network of the corpus. By contrast, WTTM values the relationship of each pattern through the word triangles and excludes some weak related ones. We conducted experiments and the results proved that WTTM did better than other baseline models. Considering the outstanding performance of WTTM, We can say that WTTM is a good choice for topic inferring on short texts.

However, there are still a lot of improvements we can do to perfect the model. How to make use of the weight of triangles can be the next point to improve. Applying our method to real-world situations is also a good direction for us to explore.

Acknowledgments. This paper is supported by the National Key Research and Development Program of China (Grant No. 2016YFB1001102) and the National Natural Science Foundation of China (Grant No. 61375069, 61403156, 61502227), this research is supported by the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing University.

References

1. Blei, D.M.: Probabilistic topic models. Commun. ACM **55**(4), 77–84 (2012)
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**(Jan), 993–1022 (2003)
3. Durak, N., Pinar, A., Kolda, T.G., Seshadhri, C.: Degree relations of triangles in real-world networks and graph models. In: Proceedings of the 21st ACM International Conference on Information and knowledge Management, pp. 1712–1716. ACM (2012)
4. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proc. Nat. Acad. Sci. **101**(suppl. 1), 5228–5235 (2004)
5. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 50–57. ACM (1999)
6. Hong, L., Davison, B.D.: Empirical study of topic modeling in twitter. In: Proceedings of the First Workshop on Social Media Analytics, pp. 80–88. ACM (2010)
7. Lu, H.Y., Xie, L.Y., Kang, N., Wang, C.J., Xie, J.Y.: Don't forget the quantifiable relationship between words: using recurrent neural network for short text topic discovery. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
8. Mimno, D., Wallach, H.M., Talley, E., Leenders, M., McCallum, A.: Optimizing semantic coherence in topic models. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 262–272. Association for Computational Linguistics (2011)
9. Weng, J., Lim, E.P., Jiang, J., He, Q.: Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, pp. 261–270. ACM (2010)
10. Yan, X., Guo, J., Lan, Y., Cheng, X.: A biterm topic model for short texts. In: Proceedings of the 22nd International Conference on World Wide Web, pp. 1445–1456. ACM (2013)
11. Zuo, Y., Zhao, J., Xu, K.: Word network topic model: a simple but general solution for short and imbalanced texts. Knowl. Inf. Syst. **48**(2), 379–398 (2016)