A Wikipedia Based Semantic Graph Model for Topic Tracking in Blogosphere

Jintao Tang[1, 2], Ting Wang[1], Qin Lu[2], Ji Wang[3], Wenjie Li[2]
[1]College of Computer, National University of Defense Technology, Changsha, P.R. China
[2]Department of Computing, Hong Kong Polytechnic University, Hong Kong
[3]National Laboratory for Parallel and Distributed Processing, Changsha, P.R. China
{tangjintao, tingwang, wj}@nudt.edu.cn, {csluqin,cswjli}@comp.polyu.edu.hk

Abstract

There are two key issues for information diffusion in blogosphere: (1) blog posts are usually short, noisy and contain multiple themes, (2) information diffusion through blogosphere is primarily driven by the "word-of-mouth" effect, thus making topics evolve very fast. This paper presents a novel topic tracking approach to deal with these issues by modeling a topic as a semantic graph, in which the semantic relatedness between terms are learned from Wikipedia. For a given topic/post, the name entities, Wikipedia concepts, and the semantic relatedness are extracted to generate the graph model. Noises are filtered out through the graph clustering algorithm. To handle topic evolution, the topic model is enriched by using Wikipedia as background knowledge. Furthermore, graph edit distance is used to measure the similarity between a topic and its posts. The proposed method is tested by using the real-world blog data. Experimental results show the advantage of the proposed method on tracking the topic in short, noisy texts.

1 Introduction

As one of the most popular personal diaries, Blogosphere has become a significant part of Internet social media. Millions of people present their opinions on hot topics, share interests and disseminate contents with each other. With the rapid increase of daily-published posts, how to track topics in massive blog posts becomes a challenging issue today.

Currently, the state of the art systems for tracking topics on the Internet mainly use the same technologies as topic detection and tracking (TDT) [James Allan, 2002], which is based on the "bag of words" representation and statistical supervised/semi-supervised learning algorithms. These technologies are useful in the traditional mainstream media. However, they cannot effectively handle the information spread through blogosphere as it poses some new challenges. At first, blog posts have much more noises than the traditional texts. Webpages usually contain contents irrelevant to the main topic, such as advertisements, navigation bars, etc. Secondly, blog posts are usually very short. The short text in posts cannot provide enough word co-occurrences context to measure their semantic similarity. Furthermore, the topics spread through blogosphere are primarily driven by the "word-of-mouth" effect, thus making topics evolve very fast. For a given topic, the posterior posts usually contain some new terms that do not exist in previous posts. Therefore, the ability of modeling the constantly changing topic is very important to the topic model.

This paper proposes a topic tracking approach based on a novel semantic graph model. It makes use of the external knowledge base (namely, Wikipedia) as the background to overcome the problem of semantic sparseness and topic evolution in the noisy, short blog posts. For each blog post, the name entities and Wikipedia concepts are recognized as nodes, and the edges which indicated the semantic relationship between the nodes also are extracted to generate the semantic graph representation. The noise terms and irrelevant topics in the posts are then filtered out through graph clustering algorithms. The graph edit distance (GED) is used to measure the semantic similarity between the topic and the posts. Furthermore, the topic model is dynamically updated with the tracked posts to deal with topic evolution. The main contributions of this paper are:

- The proposed topic model based on semantic information contained in Wikipedia is capable of solving the so-called synonym problem. If two posts use synonymous words to describe the same topic, it is difficult to get the semantic similarity between them using the bag-of-words representation. In Wikipedia, all the equivalent concepts are grouped together by redirected links. Therefore, the proposed model based on Wikipedia can easily map synonyms to the same concept.
- The use of a graph clustering algorithm helps to filter out the noisy, multi-theme text. Through the identification of the semantic relationships between keywords, the clustering algorithm helps to capture the main topic of a post.
- The proposed topic model using Wikipedia is also naturally adaptive to model topic evolution because the semantic relatedness is modeled.
- To avoid the need for training data for statistical information, we simply use graph similarity measures based on graph edit distance to evaluate the semantic similarity between the topic model and the posts.

2 Related works

Information diffusion tracking/modeling on the Internet has already attracted intensive researches. Most studies are conducted on traditional news stream, such as the topic detection and tracking [James Allan, 2002]. Recently, researchers put more focus on information diffusion in social media. Existing studies have focused on modeling the "word-of-mouth" effect among social network members [Domingos *et al*., 2001]. [Gruhl *et al*., 2004] collected a number of blog posts, and used the innovation propagation theory to study the law of information diffusion in these blog posts. Tracking information diffusion in blog has also been studied by [Leskovec *et al*., 2007] and [Adar *et al*., 2005], with or without the use of social networks.

The aforementioned methods are based on the bag-of-words representations such as TF-IDF, LDA, which are not suitable to handle blog posts. There are insufficient co-occurrences between keywords in short posts to generate an appropriate word vector representation. There are already some text mining methods using background knowledge represented by ontology, such as WordNet [Hotho *et al*., 2003], Mesh [Zhang *et al*., 2007] and so on. But these ontologies are not quite suitable for comprehensively covering of needed semantic information in blog texts. Recently, there are some researches to make use of Wikipedia to enhance text mining, such as text classification [Phan *et al*., 2008], ontology construction [Cui *et al*., 2009] and key terms extraction [Grineva *et al*., 2009]. And a few of good works which use Wikipedia to compute semantic relatedness also are emerged. These works take into consideration the links within the corresponding Wikipedia articles [Turdakov *et al*., 2008], the categories structure [Strube *et al*., 2006], or the article's textual content [Gabrilovich *et al*., 2007]. In this paper, we adopt the method proposed in [Milne *et al*., 2008]. However, this method is still limited by the coverage of Wikipedia thesaurus and will be modified in this work.

3 The Semantic Graph Model

3.1 Semantic Graph Representation

Generally speaking, the topic tracking task in blogosphere is to track a given topic (usually a new hot event) in a collection of posts according to their published time. The most popular method is to reduce each document in the corpus to a vector of real numbers, each of which represents the ratio of word occurrences. To represent the semantics in the text, there are a lot of recent works using some external resources. However, many emerging topics in blogosphere are likely to be related to some new entities. The resources like Wikipedia may not have the most up-to-date information of all the entities. For instance, the Wikipedia terms "Floods" used in a document cannot distinguish the new event "Southern African Flooding in Dec 2010" from other floods events.

We consider it appropriate to represent a topic by the name entities, concepts and the semantic relatedness between them, where the concepts indicate the kind of the event and the name entities identify the specific events, such as where the event happened or who were involved in it. According to this idea, we propose a semantic graph model, called the Entity-Concept Graph (hereafter ECG), which contains two types of nodes, the Wikipedia concept nodes and the name entity nodes, to model the general conceptual aspect of an event as well as the specificity aspect of an event. The edge weights to indicate the semantic relatedness between the two types are also different; the semantic relatedness between two concepts is learned from Wikipedia categories/links structure, and the semantic relatedness between the name entities and concepts is measured based on the co-occurrences. Figure 1 show an illustration of the ECG model about the event "Southern Africa flooding".

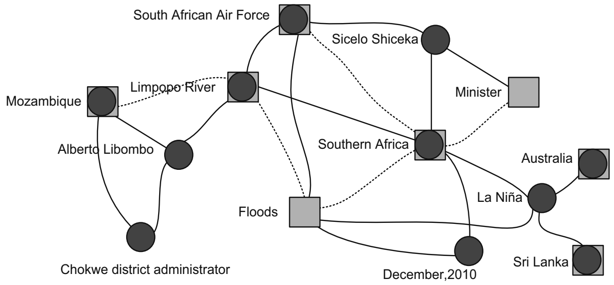

Figure 1. An example of the ECG Representation with square nodes for Wikipedia terms and circle nodes for name entities.

For name entity recognition, the well-known NER toolkit published by [Finkel *et al*., 2005] is used. To recognize the Wikipedia concepts, the Wikipedia miner toolkit proposed in [Milne *et al*., 2008] is used. We also use this toolkit to resolve the synonymy and polysemy problem based on the redirect links in Wikipedia.

ECG is defined as an undirected weighted graph, in which nodes are used to represent the keywords and the weight of an edge indicates the strength of the semantic relatedness between them. The semantic relatedness between Wikipedia terms is measured by using the link structures found within the corresponding articles in Wikipedia. The nodes in ECG stand for both name entities and Wikipedia terms. However, some name entities are not in the Wikipedia thesaurus. Thus, the semantic relatedness among these name entities cannot be inferred from Wikipedia. It is reasonable to assume that the semantic relatedness inferred from Wikipedia indicates the semantic relatedness as background global knowledge, and the co-occurrences indicate the relationship in the specific document. So we take both the semantic relatedness and the co-occurrences information to measure the relationship between key terms. If the pairs of keywords are both in the Wikipedia thesaurus, we simply use the semantic relatedness extracted from Wikipedia as the edge weight in ECG. Moreover, for any pairs of keywords, the paragraph level co-occurrences of them in all blog posts are also counted and normalized. Then, the weight of an edge is defined as follow:

$$w(e) = \begin{cases} co(v_1, v_2), v_1 \notin Wiki \text{ or } v_2 \notin Wiki \\ \alpha \cdot co(v_1, v_2) + \beta \cdot related(v_1, v_2), otherwise \end{cases} \quad (1)$$

where $co(v_1, v_2)$ is the normalized co-occurrence value of v_1,

v_2, $related(v_1,v_2)$ is the semantic relatedness value of two Wikipedia terms. $\alpha=0.3$ and $\beta=0.7$ is set on our experiences.

3.2 Filtering the Noise

Webpages used for Blog posts are usually multi-theme containing the navigation bar, advertisements, archive posts besides the main article. Fortunately, the ECG model can easily identify the main topic in the noisy blog post by using a graph-clustering algorithm.

At first, the edges with a small weight are considered as unimportant relations and removed. The isolated nodes in ECG, which are considered the noisy terms in the document, are also removed. We then use the Girvan-Newman algorithm [Girvan *et al.*, 2002] to discover the clusters in ECG. The goal of the clustering algorithm is to find the main topic in the multi-topics document. We rank the importance of the discovered clusters and select the cluster with the highest value as the main topic of the document. The ranking method is the same as that in [Grineva *et al.*, 2009].

4 Topic Tracking

This section describes the topic tracking method used in ECG. Unlike the traditional similarity measures such as the cosine distance, we measure the similarity between a topic and a post by computing the graph edit distance between their ECGs. This method can utilize Wikipedia as background knowledge straightforwardly.

4.1 Graph Edit Distance

The similarity between graphs can be considered as an inexact graph-matching problem. As a base for inexact graph matching, the graph edit distance (GED) [Gao *et al.*, 2010] measures the similarity between pairwise graphs error tolerantly. GED is defined as the cost of the least expensive sequence of edit operations to transform one graph to another. Let G_1 and G_2 be the two graphs. Let $C=(c_{nd}(v),c_{ni}(v),c_{ns}(v_1,v_2),c_{ed}(e),c_{ei}(e),c_{es}(e_1,e_2))$ be the cost function vector, where each component in C represents a cost for node deletion, node insertion, node substitution, edge deletion, edge insertion, and edge substitution, respectively. Then the edit distance is defined as follow:

$$dis(G_1,G_2)=min(C(\S)) \qquad (2)$$

where $\S$ is an error-tolerantly graph matching from G_1 to G_2.

The definition of the cost functions is the key to the GED measure. According to the research in [Bunke, 1997], if $c_{ns}(v_1,v_2)>c_{nd}(v_1)+c_{ni}(v_2)$ and $c_{es}(e_1,e_2)>c_{ed}(e_1)+c_{ei}(e_2)$, the graph edit distance from G_1 to G_2 can be represented by the size of the maximum common sub-graph and the minimum common super-graph. To be more specific, if the costs function $C=\{c,c,2c,c,c,2c\}$, the GED from G_1 to G_2 can be computed as follow:

$$dis(G_1,G_2)= c(|\tilde{G}|-|\hat{G}|) \qquad (3)$$

where $\tilde{G}$ is the minimum common super-graph of G_1 and G_2, $\hat{G}$ is the corresponding maximum common sub-graph.

4.2 Similarity Measure

The graph edit distance is an effective way to measure the similarity between two graphs. However, measuring the similarity between the topic model and the posts has a number of differences from the graph-matching problem. Most of the time a blog post simply describes a small snippet of a topic. If a new post is considered to belong to a given topic, the post ECG should be a sub-graph of the topic model. If two graphs, G_1 and G_2, both are sub-graphs of the topic model G_T, the GED value of transforming G_T to the two graphs should be the same. The cost of deleting the nodes and edges in G_T-$G_{p.}$ should be assigned to 0.

Furthermore, according to the topic evolution phenomenon, the posterior posts may contain some new terms. A few new terms are introduced by the topic evolution and others are more likely to be noise only. How to separate the semantic related new terms from the noise is critical for topic tracking in blogosphere. In this paper, we make use information in Wikipedia to construct a background graph to help measure whether a new term is semantically relevant to the topic. We can learn the semantic relatedness among all the Wikipedia concepts based on the category/link structure in Wikipedia to generate the background graph. If the new term is a Wikipedia concept, the closeness value of the new term to all the Wikipedia concepts in the topic model is computed by using the background graph, which can be considered as the cost of insert the new term into the topic model. If the new term is an entity, the closeness between the new term and the topic model is computed based on the co-occurrences in the posts corpora.

So, we redefine the cost function of the node/edge insertion, deletion and substitution for transforming the topic ECG to the post ECG. Let $G_T=(V_t, E_t)$ be the topic model ECG. *Let* $G_P=(V_p, E_p)$ be the ECG of a given post. Let $G_w=(V_w, E_w)$ be the background graph based on Wikipedia, the cost functions C are defined as follows:

$$c_{nd}(v)= c_{ed}(e)=0 \qquad (4)$$

$$c_{ns}(v_1,v_2)= c_{es}(e_1,e_2)=2 \qquad (5)$$

$$c_{ni}(v)=\begin{cases} 1-\sum_{v_k\in V_t\cap V_w} related(v,v_k) \Big/ k, \text{ if } \mathrm{v}\in V_w \\ \sum_{v_i\in V_t} co(v,v_i) \Big/ i, otherwise \end{cases} \qquad (6)$$

$$c_{ei}(e)=1-w(e) \qquad (7)$$

where $co(v_1,v_2)$, $related(v_1,v_2)$ and $w(e)$ is defined in *(1)*.

Finally, the method introduced in [Bunke, 1997] is used to compute the GED value efficiently.

4.3 Updating Topic Model

If a post is considered to belong to a given topic according to the graph edit distance measure, the topic model should be updated dynamically to represent the possible topic evolution. We update the topic model by computing the minimum common super-graph of the topic model and the ECG of the relevant post. Then the filtering step is applied in the common super-graph to remove the semantic unrelated key terms. Let G_T be the ECG of the topic model, G_P be the ECG of a post which is relevant to the topic. The main steps of the update algorithm are described as follow:

(a). Generate a new empty graph $G_N=(V_N, E_N)$, where $V_N=\{\emptyset\}$ and $E_N= \{\emptyset\}$.

(b). Let $V_N= V_P \cup V_T$, where V_P is the node set of topic graph G_P and V_T is the node set of G_T.

(c). For each edge $e \in E_T$, add e to E_N, if $e \in E_P$, update the weight of e in E_N by the formula:

$$W_N(e)= 0.9W_P(e)+ 0.1W_T(e)$$

(d). Filter the unimportant edges and isolated nodes in G_N using the method described in section 3.3. The resulting G_N is used as the new topic model.

5 Experiments

5.1 Dataset and Experiment Settings

Since there is no standard benchmark for information diffusion tracking in blogosphere, we have collected a real dataset from Internet by querying the Google blog search engine. We take the *2010 Yushu earthquake* struck on April 14 as the event. So we use the keywords of the topic ("yushu earthquake") to launch a query search and retrieved the top 100 relevant posts in Google in April 16, 2010. To generate the dataset for information diffusion tracking, we subscribed the search results in Google and retrieved the daily updates. As a result, 652 posts are obtained from the RSS feed since April 14 to May 2. We also crawled 4,000 other posts obtained from the Google blog RSS feeds including politics, news and sports to admixture the dataset.

To initialize the topic model, we have selected the earliest 10 relevant blog posts based on the timestamps. And then the proposed tracking method is applied to the data chronically. The search results of "yushu earthquake" from Google in May 2 are used as the baseline. For comparison, two widely used methods to track the topic diffusion are also implemented and evaluated. One is a word vector model which chooses keywords based on the TF-IDF score and uses the cosine distance to measure the similarity (hereafter TF-IDF). The other considers the topic-tracking problem as a classification task, which uses the LDA model as document representation to train the SVM model to classify the posts as either topic relevant or irrelevant [Blei *et al.*, 2002] (hereafter LDA-SVM).

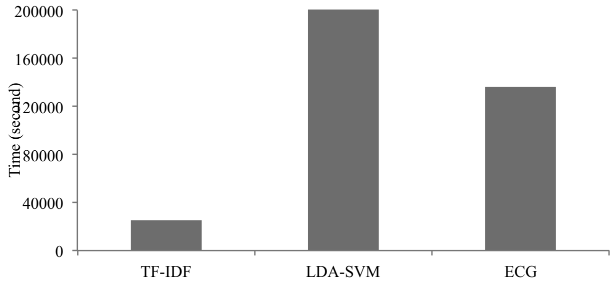

Figure 2. The run time of tracking methods when tracking the given topic in 4652 blog posts

Figure 2 shows the run time performance of our proposed ECG compared to the TF-IDF and LDA-SVM. To our surprise, LDA-SVM takes 25% more time to run compared to the ECG model based method. Although the complexity of graph edit distance computing is high, the graph representation of the post and the topic model usually are small according to the noise filtering. Training the LDA model and SVM classifier are time consuming, so the performance of LDA-SVM is worse than the proposed method.

5.2 Precision and Recall

We randomly select 500 posts as the test dataset. Five volunteers are asked to manually annotate whether the posts are related to the topic. Each volunteer is asked to annotate 300 posts and each post is separately judged by 3 volunteers to mark whether the posts are related to the topic. The average inter-annotator agreement is 85.2% for 3 annotators. Then the precision, recall and F-measure are used to evaluate the aforementioned methods.

Table 1. The precision, recall and F-measure of topic tracking.

	Precision	Recall	F Measure
Google	0.804	0.831	0.818
TF-IDF	0.795	0.787	0.791
LDA-SVM	**0.857**	0.801	0.828
ECG (GED<5)	0.844	0.854	0.849
ECG (GED<10)	0.842	0.898	**0.870**
ECG (GED<15)	0.786	**0.910**	0.844

Table 1 shows that the proposed tracking method with an appropriate similarity threshold (GED<10) has the best F1 score among all the methods in the experiment. The TF-IDF based method shows the poorest performance, even worse than the search results of Google. The bag-of-words model, due to its inability in semantic representation, cannot distinguish the important terms from the noise effectively. The LDA-SVM gets the best precision in the experiments. However, unlike previous experiments on the news datasets Reuters-21578 [Blei *et al.*, 2002], the LDA based method did not show the advantage on recall when applied to the noisy, short blog data. The lack of background knowledge probably exerts a negative influence on the recall of the LDA based method. Different from the previous methods, the ECG model performs well on blog data. The ECG based method with different thresholds archive better F1 score than that of the LDA-SVM. This proves that considering pairwise semantic relatedness between key terms based on a global background knowledge base (such as Wikipedia) can improve the performance of tracking topics in noisy webpages. The best F1 result of the ECG is 0.870, which are 4% higher than that of the LDA-SVM.

As shown in Table 1, the definition of GED threshold is a key issue to the effectiveness of the ECG based method. The scalability of the ECG model for dealing with topic evolution also depends on the threshold value. We tested the influence of the GED threshold on the effectiveness of topic tracking. Figure 3 shows the curve of the precision, recall and F-measure with different thresholds in the test dataset.

The result curves indicate that the recall value increases with the GED threshold, while the precision is just the opposite. If the acceptance condition is very strict, more than 85 percent of the tracked posts really belong to the topic, and more than 75 percent of the relevant posts are discovered. It is interesting to see that the recall increases rapidly while the precision decreases only slightly when easing the threshold restriction. This phenomenon indicates that the ECG model using Wikipedia is effective to deal with topics evolution. If the threshold is set to 10, our system gets the best balance between precision and recall. Then the precision has a distinct decrease with the threshold eased, which indicates that the precision is more sensitive than recall when threshold changes. So it is better to select a low threshold to track the topic using the ECG model.

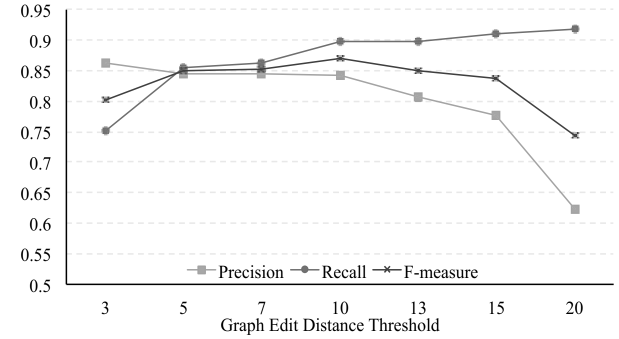

Figure 3. The curve of precision, recall and F-measure with the different graph edit distance threshold

5.3 Topic evolving Scalability

To evaluate the scalability of the proposed method on topic evolution, we compared the difference of the node sets between the initial topic model and the final topic model. Figure 4 shows the corresponding keywords in the topic model depend on occurrences and the size of a word is proportional to its frequency in all the topic-related posts. The changing of the keywords from the initial topic model to the final topic model indicates the topic evolution. As shown in Figure 4 (a), in the beginning the blog users were discussing the topic about the disaster itself, such as the time, location and so on. Figure 4 (b)&(c) shows the final topic model with different similarity thresholds, and both indicate that after a few days the discussed sub-topics are shifted to other relevant news. From Figure 4(b)&(c) we can spot some different sub topics: *the rescue team*, *Buddhist* and *monastery*, *donation* and so on. Obviously these topics are semantic related to the topic of *Yushu earthquake*. This demonstrates the ability of our proposed method on tracking the evolving topics. This ability primarily comes from the use of Wikipedia as background knowledge, which can make sure that the tracked topics are semantic related to the main topic.

5.4 Noise Stability

Usually the webpages including blog posts are very noisy. They contain many texts that are irrelevant to the main topic. The noise stability of the tracking algorithms is a key issue to the effectiveness. To evaluate the noise stability of the proposed method, we have compared the tracking results of our methods with and without the noise-filtering algorithm.

Table 2. The influence of noise filtering

	Precision	**Recall**	**F Measure**
GED<5	**0.844**	**0.854**	**0.849**
GED<5, not filtering	0.708	0.764	0.735
GED<10	**0.842**	**0.898**	**0.870**
GED<10,not filtering	0.743	0.843	0.789

Table 2 shows that noise filtering by using the graph clustering algorithm is effective for topic tracking in blog dataset. It can be observed that the tracking method has more than 5 percent lower F1 score without the filtering step, which indicates that the filtering method can distinguish the related keywords from the noises. In order to investigate the performance of the noise filtering method, we have collected the removed words with the highest term-frequency (TF) in posts to manually evaluate whether these removed words are semantic related with the topic.

Table 3. The first 10 terms ordering by the TF value which are judged as noise in the posts

	Filtered noise words
GED<5	Blogger, new York, facebook, WordPress, blog archive, Reuters, Google, Wednesday, Flickr, Google AdSense
GED<10	Blogger, newyork, BBC, facebook, blog archive, US, google, Photo gallery, WordPress, Picasa

Table 3 summarizes the first 10 removed key terms ranked by the term-frequency in blog posts. As shown in Table 3, most of the removed terms are the common concepts/name entities, such as *blogger*, *newyork* and so on. And we can learn from Table 3 that the first 10 removed terms are actually semantically unrelated to the topic *Yushu earthquake*. Although the ECG based methods with different similarity thresholds have discovered different sub topics as shown in Figure 4, the noise terms they removed are almost the same. The reason for this tendency is that noise terms are usually introduced by themes, such as the navigation bar and advertisements. The noise terms are very similar in each blog post, and exhibit low semantic relatedness with the main topic. So the proposed method can filter them out by clustering the ECG representation of blog posts.

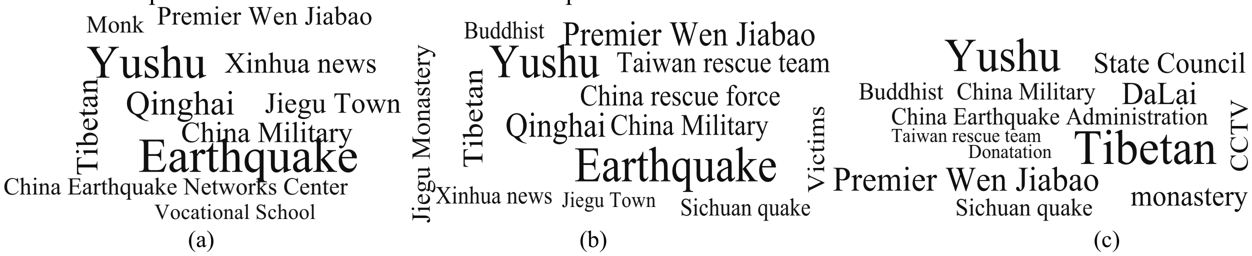

Figure 4. The key terms in the topic graph. The size of a word is proportional to its frequency in posts. (a) the initialized topic model. (b) the tracking model with threshold value<5 (c) the tracking model with threshold value<10

7 Conclusion

In this paper, we have presented a novel information diffusion tracking method based on the semantic graph topic model using Wikipedia. One of the advantages of the proposed method is that it does not require any training, while the semantic relatedness between entities and concepts is extracted upon the Wikipedia knowledge base. The important and novel feature of our method is the graph-based ECG model that can meet the challenge of tracking dynamically evolving topic in noisy, short and multi-theme blog posts. Based on the ECG representation, our method can easily identify the main topic in the blog posts and further remove noise by using a graph clustering algorithm. The graph edit distance is effective to measure the similarity between the post graph and the topic model. The incorporation of Wikipedia into the topic model is very useful to measure whether the shifted topics are semantically deviated from the tracking topic.

Future works include testing our method on greater dataset and improving the use of Wikipedia information based semantic graph model to track the topic in variety of domains.

Acknowledgments

The research is supported by the National Natural Science Foundation of China (60873097, 60933005) and the National Grand Fundamental Research Program of China (2011CB302603).

References

[Adar *et al.*, 2005] E. Adar and L. A. Adamic. Tracking information epidemics in Blogspace. In *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, 207-214, September 2005.

[Blei *et al.*, 2002] David M Blei, Andrew Y Ng, Michael I Jordan et al. Latent Dirichlet allocation. *Journal of Machine Learning Research*, Vol.3: 993-102, 2002.

[Bunke, 1997] H. Bunke. On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18(8): 689–694, 1997.

[Cui *et al.*, 2009] Gaoying Cui, Qin Lu, Wenjie Li, Yirong Chen. Mining concepts from Wikipedia for Ontology construction. In *Proceedings of the IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent*. Vol.3: 287-290, 2009.

[Domingos et al., 2001] P. Domingos, M. Richardson, Mining the Network Value of Customers. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, 57-66, 2001

[Finkel *et al.*, 2005] J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, 2005, 363-370.

[Gabrilovich *et al.*, 2007] E. Gabrilovich and S. Markovitch. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of The 20th International Joint Conference for Artificial Intelligence*, 1606–1611, Hyderabad, India, 2007.

[Gao *et al.*, 2010] X Gao, B Xiao, D Tao, X Li. A survey of graph edit distance. *Pattern Analysis & Applications*, Vol.13(1): 113-129, 2010.

[Girvan *et al.*, 2002] M Girvan, MEJ Newman. Community structure in social and biological networks. *Natural Academy Science USA*, 99:7821-7826, 2002.

[Grineva *et al.*, 2009] M. Grineva, et al. Extracting key terms from noisy and multi-theme documents. In *Proceeding of the 18th International World Wide Web Conference*, 661-670, 2009.

[Gruhl *et al.*, 2004] D. Gruhl, R. V. Guha, D. Liben-Nowell, et al. Information diffusion through blogspace. *SIGKDD Explorations*, Vol. 6(2): 43–52, 2004.

[Hotho *et al.*, 2003] A. Hotho, S. Staab, and G. Stumme. Wordnet improves text document clustering. In *Proceedings of Semantic Web Workshop, the 26th annual International ACM SIGIR Conference*, Canada, 2003.

[James Allan, 2002] James Allan. Introduction to topic detection and tracking. In *Topic detection and tracking*, James Allan (Ed.). Kluwer Academic Publishers, Norwell, MA, USA 1-16.

[Leskovec *et al.*, 2007] J. Leskovec, A Krause, et al. Cost-effective outbreak detection in networks. In *Proceedings of the 13th ACM International Conference on Knowledge Discovery and Data Mining*, 420-429, 2007.

[Milne *et al.*, 2008] D. Milne, and I.H Witten. Learning to link with Wikipedia. In *Proceedings of the ACM Conference on Information and Knowledge Management*, 509-518, California, 2008.

[Phan *et al.*, 2008] X. Phan, L. Nguyen. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceeding of the 17th international conference on World Wide Web*, 91-100, 2008.

[Strube *et al.*, 2006] M. Strube and S. Ponzetto. WikiRelate! Computing semantic relatedness using Wikipedia. In *Proceedings of the 21st National Conference on Artificial Intelligence*, 1419-1424, Boston, July 2006.

[Turdakov *et al.*, 2008] D. Turdakov and P. Velikhov. Semantic relatedness metric for Wikipedia concepts based on link analysis and its application to word sense disambiguation. In *Colloquium on Databases and Information Systems (SYRCoDIS)*, 2008.

[Zhang *et al.*, 2007] X. Zhang, L. Jing, X. Hu, et al. A comparative study of Ontology based term similarity measures on document clustering. In *Proceedings of 12th International Conference on Database Systems for Advanced Applications*. Thailand, 115-126, 2007.