Probabilistic Topic Modelling with Semantic Graph

Long Chen(✉), Joemon M. Jose, Haitao Yu, Fajie Yuan, and Huaizhi Zhang

School of Computing Science, University of Glasgow,
Sir Alwyns Building, Glasgow, UK
long.chen@glasgow.ac.uk

Abstract. In this paper we propose a novel framework, topic model with semantic graph (TMSG), which couples topic model with the rich knowledge from DBpedia. To begin with, we extract the disambiguated entities from the document collection using a document entity linking system, i.e., DBpedia Spotlight, from which two types of entity graphs are created from DBpedia to capture local and global contextual knowledge, respectively. Given the semantic graph representation of the documents, we propagate the inherent topic-document distribution with the disambiguated entities of the semantic graphs. Experiments conducted on two real-world datasets show that TMSG can significantly outperform the state-of-the-art techniques, namely, author-topic Model (ATM) and topic model with biased propagation (TMBP).

Keywords: Topic model · Semantic graph · DBpedia

1 Introduction

Topic models, such as Probabilistic Latent Semantic Analysis (PLSA) [7] and Latent Dirichlet Analysis (LDA) [2], have been remarkably successful in analyzing textual content. Specifically, each document in a document collection is represented as random mixtures over latent topics, where each topic is characterized by a distribution over words. Such a paradigm is widely applied in various areas of text mining. In view of the fact that the information used by these models are limited to document collection itself, some recent progress have been made on incorporating external resources, such as time [8], geographic location [12], and authorship [15], into topic models.

Different from previous studies, we attempt to incorporate semantic knowledge into topic models. Exploring the semantic structure underlying the surface text can be expected to yield better models in terms of their discovered latent topics and performance on prediction tasks (e.g., document clustering). For instance, by applying knowledge-rich approaches (cf. Sect. 3.2) on two news articles, Fig. 1 presents a piece of global semantic graph. One can easily see that "United States" is the central entity (i.e., people, places, events, concepts, etc. in DBPedia) of these two documents with a large number of adjacent entities. It is also clear that a given entity only have a few semantic usage (connection to other

N. Ferro et al. (Eds.): ECIR 2016, LNCS 9626, pp. 240–251, 2016.
DOI: 10.1007/978-3-319-30671-1_18

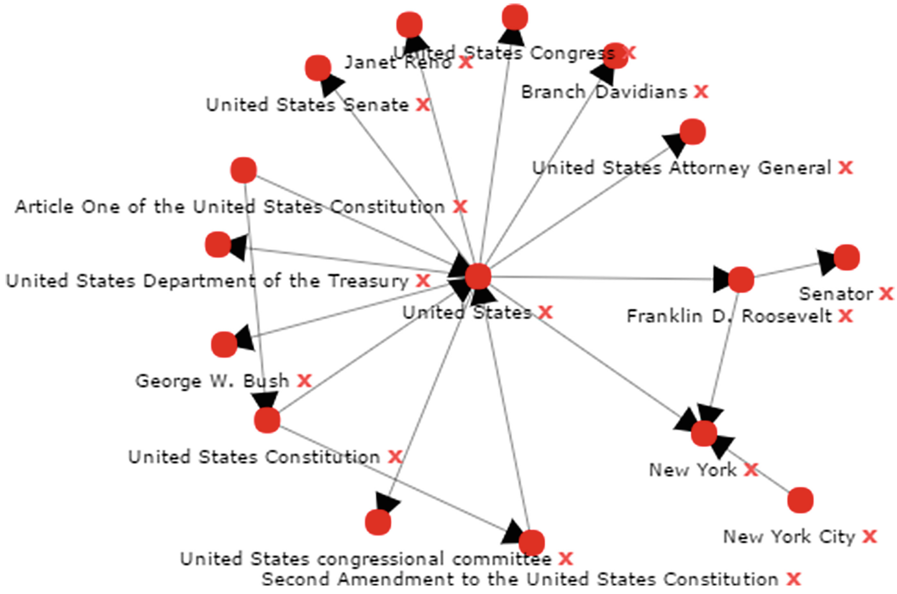
Janet Reno
United States Congress
Branch Davidians
United States Senate
United States Attorney General
Article One of the United States Constitution
United States Department of the Treasury
United States
Senator
Franklin D. Roosevelt
George W. Bush
United States Constitution
New York
New York City
United States congressional committee
Second Amendment to the United States Constitution

Fig. 1. A piece of global semantic graph automatically generated from two documents (178382.txt and 178908.txt of 20 Newsgroups dataset)

entities) and thus can only concentrate on a subset of topics, and utilization of this information can help infer the topics associated with each of the document in the collections. Hence, it is interesting to learn the interrelationships between entities in the global semantic graph, which allows an effective sharing of information from multiple documents. In addition to the global semantic graph, the inference of topics associated with a single document is also influenced by other documents that have the same or similar semantic graphs. For example, if two documents overlapped with their entities list, then it is highly possible that these two documents also have a common subset of topics. Following this intuition, we also construct local semantic graphs for each document in the collection with the hope to utilize their semantic similarity.

In a nutshell, the contribution of this paper are:

1. We investigate two types of graph-based representations of documents to capture local and global contextual knowledge, respectively, for enriching topic modelling with semantic knowledge.
2. We present a topic modelling framework, namely, Topic Models with Semantic Graph (TMSG), which can identify and exploit semantic relations from the knowledge repositories (DBpedia).
3. The experimental results on two real-world datasets show that our model is effective and can outperform state-of-the-art techniques, namely, author-topic Model (ATM) and topic model with biased propagation (TMBP).

2 Related Work

2.1 Topic Model with Network Analysis

Topic Model, such as PLSA [7] and LDA [16], provides an elegant mathematical model to analyze large volumes of unlabeled text. Recently, a large number of studies, such as Author-Topic Model (ATM) [15] and CFTM [4] have been reported for integrating network information into topic model, but they mostly focus on homogeneous networks, and consequently, the information of heterogeneous network is either discarded or only indirectly introduced. Besides, the concept of graph-based regularizer is related to Mei's seminal work [13] which incorporates a homogeneous network into statistic topic model to overcome the overfitting problem. The most similar work to ours is proposed by Deng et al. [5], which utilised the Probabilistic Latent Semantic Analysis (PLSA) [7] (cf. Sect. 3.1) together with the information learned from a heterogeneous network. But it was originally designed for academic networks, and thus didn't utilize the context information from any knowledge repository. In addition, their framework only incorporates the heterogeneous network (i.e., relations between document and entity), while the homogeneous network (i.e., relations between entity pairs with weight) is completely ignored, whereas we consider both of them in our framework.

2.2 Knowledge Rich Representations

The recent advances in knowledge-rich approaches (i.e., DBPedia[1] and Knowledge Graph[2]) provide new opportunities to gain insight into the semantic structure of a document collection. Although recent studies have already shown the effectiveness of knowledge-rich approaches in several NLP tasks such as document similarity [14], topic labelling [9], and question answering [3], its feasibility and effectiveness in topic modelling framework is mostly unknown. Hulpus et al. [9] reported a framework which extracts sub-graphs from DBpedia for labelling the topics obtained from a topic model. However, their graph construction process is relied on a small set of manually selected DBpedia relations, which does not scale and needs to be tuned each time given a different knowledge repository. Instead, we extend our semantic graphs by weighting the edges (see Sect. 3.2), which is similar to the spirit of [14]. However, there is a stark difference between their work and ours: the motivation of their work is to produce graph-representation of documents for the task of document ranking, while we aim to construct semantic graph for the task of topic modelling and documents clustering.

More generally, several semantic approaches [6,11] have been proposed to combine topic modelling with word's external knowledge. However, they either relied on a small-scale semantic lexicon, e.g., WordNet, or didn't consider the relationship of entities. In contrast, we used a larger widely-covered ontology with a general-purpose algorithm to propagate the inherent topic-entity distribution.

[1] http://wiki.dbpedia.org/.

[2] https://developers.google.com/freebase/.

3 Models

3.1 Probabilistic Topic Model

In PLSA, an unobserved topic variable $z_k \in \{z_1, ..., z_K\}$ is associated with the occurrence of a word $w_j \in \{w_1, ..., w_M\}$ in a particular document $d_i \in \{d_1, ..., d_N\}$. After the summation of variable z, the joint probability of an observed pair (d, w) can be expressed as

$$P(d_i, w_j) = P(d_i) \sum_{k=1}^{K} P(w_j|z_k)P(z_k|d_i) \tag{1}$$

where $P(w_j|z_k)$ is the probability of word w_j according to the topic model z_k, and $P(z_k|d_j)$ is the probability of topic z_k for document d_i. Following the likelihood principle, these parameters can be determined by maximizing the log likelihood of a collection C as follows:

$$L(C) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) log \sum_{k=1}^{K} P(w_j|z_k)P(z_k|d_i) \tag{2}$$

The model parameters $\phi = P(w_j|z_k)$ and $\theta = P(z_k|d_i)$ can be estimated by using standard EM algorithm [7].

Thus PLSA provides a good solution to find topics of documents in a text-rich information network. However, this model ignores the associated heterogeneous information network as well as other interacted objects. Furthermore, in PLSA there is no constraint on the parameters $\theta = P(z_k|d_i)$, the number of which grows linearly with the data. Therefore, the model tends to overfitting the data. To overcome these problems, we propose to use a biased propagation algorithm by exploiting a semantic network.

3.2 Topic Modelling with Semantic Graph

In this section, we propose a biased propagation algorithm to incorporate the entity semantic network with the textual information for topic modelling, so as to estimate the probabilities of topics for documents as well as other associated entities, and consequently improve the performance of topic modelling. Given the topic probability of documents $P(z_k|d_i)$, the topic probability of an entity can be calculated by:

$$\begin{aligned} P(z_k|e) &= \frac{1}{2}(\sum_{d_i \in D_e} P(z_k|d_i)P(d_i|e) + \sum_{e_j \in C_e} P(z_k|e_j)P(e_j|e)) \\ &= \frac{1}{2}(\sum_{d_i \in D_e} \frac{P(z_k|d_i)}{|D_e|} + \sum_{e_j \in C_e} P(z_k|e_j)P(e_j|e)) \end{aligned} \tag{3}$$

where D_e is a set of documents that contain the entity e, C_e is a set of entities which are connected to entity e. $P(z_k|e_j)$ is the topic probability of entity e_j, which is estimated with a similar manner as $P(z_k|d_i)$ by using the EM algorithm (see Sect. 3.3). $P(e_j|e)$ is the highest weight between entity e_j and e (see Sect. 3.2). The underlying intuition behind the above equation is that the topic distribution of an entity is determined by the average topic distribution of connected documents as well as the connected entities of semantic graph. On the other hand, the topic distributions could be propagated from entities to documents, so as to reinforce the topic distribution of documents. Thus we propose the following topic-document propagation based on semantic graph:

$$P_E(z_k|d) = \xi P(z_k|d) + (1-\xi) \sum_{e \in E_d} \frac{P(z_k|e)}{|E_d|} \tag{4}$$

where E_d denotes the set of entities of document d, ξ is the biased parameter to strike the balance between inherent topic distribution $P(z_k|d)$ and entity topic distribution $P(z_k|e)$. If $\xi = 1$, the topics of documents retain the original ones. If $\xi = 0$, the topic of the documents are determined by the entity topic distribution. By replacing the $P(z_k|d)$ in $L(C)$ with $P_E(z_k|d)$ in Eq. 4, the log-likelihood of TMSG is given as:

$$L'(C) = \sum_{i=1}^{N} \sum_{j=1}^{M} n(d_i, w_j) log \sum_{k=1}^{K} P(w_j|z_k) P_E(z_k|d_i) \tag{5}$$

Semantic Graph Construction. When computing the $P(e_j|e)$ in the above, TMSG model, we adopt the method of [14] to construct the semantic graph. We start with a set of input entities C, which is found by using the off-the-shelf named entity recognition tool DBpedia Spotlight[3]. We then search a sub-graph of DBpedia which involes the entities we already identified in the document, together with all edges and intermediate entities found along all paths of maximal length L that connect them. In this work, we set $L = 2$, as we find when L is larger than 3 the model tends to produce very large graphs and introduce lots of noise.

Figure 2 illustrates an example of a semantic graph generated from the set of entities {**db:Channel, db:David Cameron, db:Ed Miliband**}, e.g. as found in the sentence "Channel 4 will host head-to-head debates between David Cameron and Ed Miliband." Starting from these seed entities, we conduct a depth-first search to add relevant intermediate entities and relations to G (e.g., **dbr:Conservative Party** or **foaf:person**). As a result, we obtain a semantic graph with additional entities and edges, which provide us with rich knowledge about the original entities. Notice that we create two versions of semantic graphs, namely, the local semantic graph and global semantic graph. The local entity graphs build a single semantic graph for each document, and it aims to capture

[3] https://github.com/dbpedia-spotlight/dbpedia-spotlight.

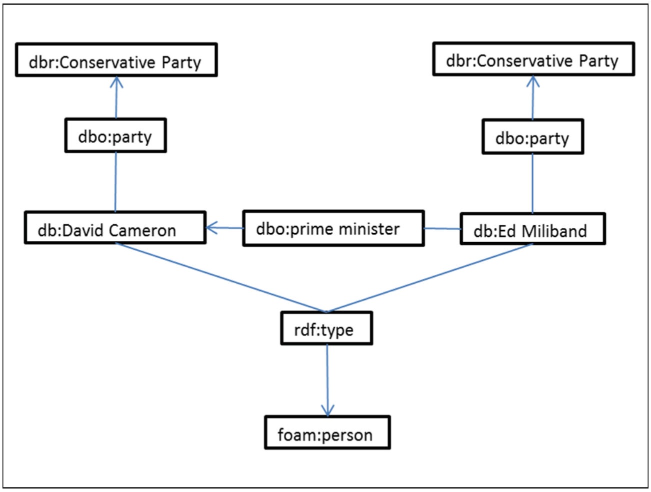

Fig. 2. A Sample Semantic Graph

the document context information. The global entity graph is constructed with the entities of the whole document collection, and we use it to detect the global context information.

Semantic Relation Weighting. So far, we simply traverse a set of input entities from DBpedia graph. However, DBpedia ontology contains many fine-grained semantic relations, which may not be equally informative. For example, in Fig. 2 seed entities **db:David Cameron** and **db:Ed Miliband** can be connected through both **rdf:type foaf:person** and **dbpprop:birthPlace**. However the former is less informative since it can apply to a large amount of entities (i.e., all persons in DBpedia). Weights can capture the degree of correlation between entities in the graph, and the core idea underlying our weighting scheme is to reward those entities and edges that are most specific to it. We define the weighting function as $W = -\log(P(W_{Pred}))$, where W is the weight of an edge, $P(W_{Pred})$ is the probability that the predicate W_{Pred} (such as rdf:type) describing the specific semantic relation. This measure is based on the hypothesis that specificity is a good proxy for relevance. We can compute the weights values for all types of predicates, as we have the whole DBpedia graph available and can query for all possible realizations of the variable X_{Pred}. We obtain the probability $P(W_{Pred})$ through maximum likelihood estimation, which is calculated by the frequency of W_{Pred} type divided by the overall counting of all the predicates.

3.3 Model Fitting with EM Algorithm

When a probabilistic model involves unobserved latent variables, the standard way is to employ the Expectation Maximization (EM) algorithm, which alternates two steps, E-step and M-step. Formally, we have the E-step to calculate the posterior probabilities $P(z_k|d_i, w_j)$ and $P(z_k|d_i, e_l)$:

$$P(z_k|d_i, w_j) = \frac{P(w_j|z_k)P_E(z_k|d_i)}{\sum_{k'=1}^{K} P(w_j|z_{k'})P_E(z_{k'}|d_i)} \tag{6}$$

$$P(z_k|d_i, e_l) = \frac{P(e_l|z_k)P_E(z_k|d_i)}{\sum_{k'=1}^{K} P(e_l|z_{k'})P_E(z_{k'}|d_i)} \tag{7}$$

In the M-step, we maximize the expected complete data log-likelihood for PLSA, which can be derived as:

$$Q_D = \sum_{i=1}^{N}\sum_{j=1}^{M} n(d_i, w_j) \sum_{k=1}^{K} P(z_k|d_i, w_j) log \sum_{k=1}^{K} P(w_j|z_k)P_E(z_k|d_i) \tag{8}$$

There is a closed-form solution [5] to maximize Q_D:

$$P(w_j|z_k) = \frac{\sum_{i=1}^{N} n(d_i, w_j)P(z_k|d_i, w_j)}{\sum_{j'=1}^{M}\sum_{i=1}^{N} n(d_i, w_{j'})P(z_k|d_i, w_{j'})} \tag{9}$$

$$\begin{aligned} P_E(z_k|d_i) = \xi \frac{\sum_{j=1}^{M} n(d_i, w_j)P(z_k|d_i, w_j)}{\sum_{j'=1}^{M} n(d_i, w_{j'})} + \\ (1-\xi)\frac{\sum_{l=1}^{K} n(d_i, e_l)P(z_k|d_i, e_l)}{\sum_{l'=1}^{K} n(d_i, e_{l'})} \end{aligned} \tag{10}$$

It is possible to employ more advanced parameter estimating methods, which is left for future work.

4 Experimental Evaluation

We conducted experiments on two real-world datasets, namely, DBLP and 20 Newsgroups. The first dataset, DBLP[4], is a collection of bibliographic information on major computer science journals and proceedings. The second dataset, 20 Newsgroups[5], is a collection of newsgroup documents, partitioned evenly across 20 different newsgroups. We experimented with topic modelling using a similar set-up as in [5]: For DBLP dataset, we select the records that belong to the following four areas: *database, data mining, information retrieval, and artificial intelligence*. For 20 Newsgroups dataset, we use the full dataset with 20 categories, such as *atheism, computer graphics*, and *computer windows X*.

For preprocessing, all the documents are lowercased and stopwords are removed using a standard list of 418 words. With the disambiguated entities (cf. 3.2), we create local and global entity collections, respectively, for constructing local and global semantic graphs. The creation process of entity collections is organized as a pipeline of filtering operations:

[4] http://www.informatik.uni-trier.de/~ley/db/.

[5] http://qwone.com/~jason/20Newsgroups/.

Table 1. Statistic of the DBLP and 20 Newsgroups datasets

	DBLP	20 Newsgroups
# of docs	40,000	20,000
# of entities (local)	89,263	48,541
# of entities (global)	9,324	8,750
# of links (local) docs	237,454	135,492
# of links (global) docs	40,719	37,713

1. The isolated entities, which have no paths with the other entities of the full entity collection in the DBpedia repository, are removed, since they have less power in the topic propagation process.
2. The infrequent entities, which appear in less than five documents when constructing the global entity collection, are discarded.
3. Similar to step 2, we discard entities that appear less than two times in the document when constructing the local entity collection.

The statistic of these two datasets along with their corresponding entities and links are shown in Table 1. We randomly split each of the dataset into a training set, a validation set, and a test set with a ratio 2:1:1. We learned the parameters in the semantic graph based topic model (TMSG) on the training set, tuned the parameters on the validation set and tested the performance of our model and other baseline models on the test set. The training set and the validation set are also used for tuning parameters in baseline models. To demonstrate the effectiveness of the TMSG method, we introduce the following methods for comparison:

- **PLSA:** The baseline approach which only employs the classic Probabilistic Latent Semantic Analysis [7].
- **ATM:** The state-of-the-art approach, Author Topic Model, which combines LDA with authorship network [15], in which authors are replaced with entities.
- **TMBP:** The state-of-the-art approach, Topic Model with Biased Propagation [5], which combines PLSA with an entity network (without the external knowledge, such as DBpedia).
- **TMSG:** The approach which described in Sect. 3, namely, Topic Model with Semantic Graph.

In order to evaluate our model and compare it to existing ones, we use accuracy (AC) and normalized mutual information (NMI) metrics, which are popular for evaluating effectiveness of clustering systems. The AC is defined as $AC = \frac{\sum_1^n \delta(a_i, map(l_i))}{n}$ [17], where n denotes the total number of documents, $\delta(x, y)$ is the delta function that equals one if $x = y$ and equals zero otherwise, and $map(l_i)$ is the mapping function that maps each cluster label l_i to the equivalent label from the data corpus. Given two set of documents, C and C', their mutual information metric $MI(C, C')$ is defined as: $MI(C, C') =$

$\sum_{c_i \in C, c'_j \in C'} p(c_i, c'_j) \cdot log_2 \frac{p(c_i, c'_j)}{p(c_i) \cdot p(c'_j)}$ [17], where $p(c_i)$ and pc'_j are the probabilities that a document arbitrarily selected from the corpus belongs to the clusters c_i and c'_j, respectively, and $p(c_i, c'_j)$ is the joint probability that arbitrarily selected document belongs to the cluster c_i and c'_j at the same time.

4.1 Experimental Results

Parameter Setting: For PLSA, we only use textual content for documents clustering with no additional entity information. For ATM, we use symmetric Dirichlet priors in the LDA estimation with $\alpha = 50/K$ and $\beta = 0.01$, which are common settings in the literature. For TMBP model, an entity-based heterogeneous network is constructed, and its parameter settings were set to be identical to [5]. Consistent to our previous setting of categories, we set the number of topics (K) to be four for DBLP and twenty for 20 Newsgroups as we need the data label for calculating the accuracy. The essential parameter in this work is ξ which controls the balance between the inherent textual information and semantic graph information (cf. Sect. 3.2). Figures 3 and 4 show how the performance varies with the bias parameter ξ. When $\xi = 1$, it is the baseline PLSA model. We see that the performance is improved over the baseline when incorporating

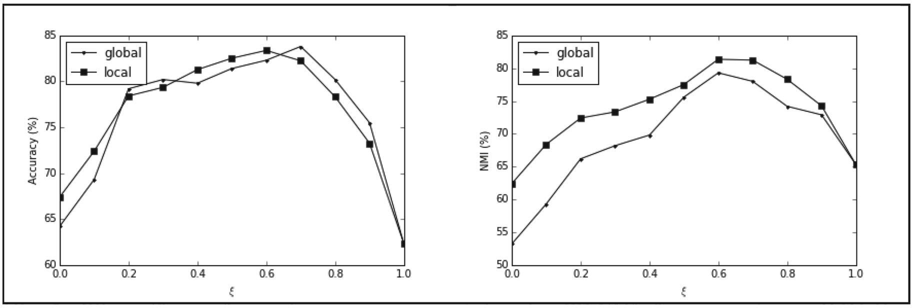

Fig. 3. The effect of varying parameter ξ in the TMSG framework on DBLP dataset.

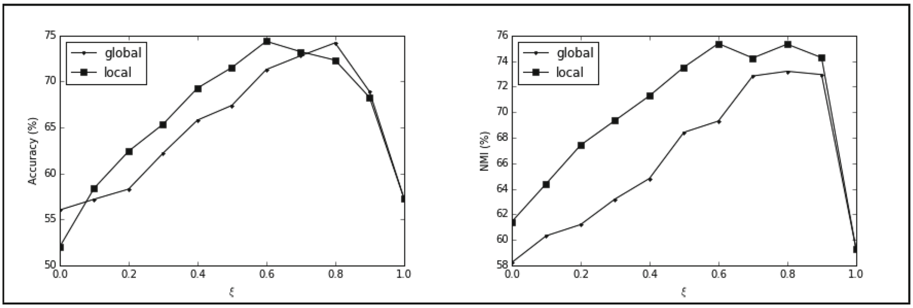

Fig. 4. The effect of varying parameter ξ in the TMSG framework on 20 Newsgroups dataset.

the semantic graph with $\xi < 0.6$. It is also notable that the TMSG with local semantic graphs (local TMSG) generally performs better then the TMSG with global semantic graph (global TMSG), which suggests that the local context is probably more important than the global one for document clustering task. We further tuned the parameters on the validation dataset. When comparing TMSG with other existing techniques, we empirically set the bias parameter $\xi = 0.6$ and the ratio between local and global TMSG is set as 0.6 : 0.4.

Table 2 depicts the clustering performance of different methods. For each method, 20 test runs are conducted, and the final performance scores were calculated by averaging the scores from the 20 tests. We can observe that ATM outperforms the baseline PLSA with additional entity network information. As expected, TMBP outperforms the ATM since it directly incorporates the heterogeneous network of the entities. More importantly, our proposed model TMSG can achieve better results than state-of-the-art ATM and TMBP algorithms. A comparison using the paired t-test is conducted for PLSA, ATM, and TMBP over TMSG, which clearly shows that our proposed TMSG outperforms all baseline methods significantly. This indicates that by considering the semantic graph information and integrating with topic modelling, TMSG can have better topic modelling power for clustering documents.

Table 2. The clustering performance of different methods on (a) DBLP and (b) 20 Newsgroups datasets (-*-* and -* indicate degraded performance compared to TMSG with p-value < 0.01 and p-value < 0.05, respectively).

(a) DBLP

	PLSA	**ATM**	**TMBP**	**TMSG**
AC	0.62-*-*	0.68-*	0.72-*	0.80
NMI	0.65-*-*	0.72-*	0.75-*	0.82

(b) 20 Newsgroups

	PLSA	**ATM**	**TMBP**	**TMSG**
AC	0.56-*-*	0.63-*-*	0.67-*	0.72
NMI	0.55-*-*	0.61-*-*	0.65-*	0.71

Table 3. The representative terms generated by PLSA, ATM, TMBP, and TMSG models. The terms are vertically ranked according to the probability $P(w|z)$.

	Topic 1 (DB)		Topic 2 (DM)		Topic 3 (IR)		Topic 4 (AI)	
PLSA	data	management	data	algorithm	information	learning	learning	knowledge
	database	processing	mining	performance	retrieval	search	algorithm	time
	memory	relational	learning	detection	web	**system**	application	logic
	system	processing	clustering	analysis	knowledge	language	human	**search**
	architecture	**feature**	classification	parameter	text	query	model	representation
ATM	data	management	mining	**multiple**	information	language	learning	algorithm
	database	software	data	algorithm	retrieval	text	knowledge	paper
	server	relational	classification	performance	search	web	logic	time
	system	function	learning	analysis	knowledge	**classification**	**image**	**method**
	query	processing	clustering	detection	**performance**	query	model	application
TMBP	data	software	data	parameter	information	learning	knowledge	paper
	database	relational	mining	algorithm	retrieval	query	application	**intelligence**
	management	architecture	classification	**result**	document	**estimation**	human	model
	algorithm	**text**	learning	analysis	query	**management**	algorithm	system
	server	processing	clustering	**time**	web	language	**compute**	performance
TMSG	data	**accelerator**	data	**analysis**	information	search	knowledge	logic
	database	function	mining	algorithm	retrieval	document	learning	system
	query	relational	classification	parameter	query	**semantic**	information	**data**
	system	software	clustering	**pattern**	knowledge	language	information	representation
	distributed	**performance**	learning	**information**	text	**user**	**reasoning**	uman

Since the DBLP dataset is a mixture of four areas, it is interesting to see whether the extracted topics could reflect this mixture. Shown in Table 3 are the most representative words of topics generated by PLSA, ATM, TMBP, and TMSG, respectively. For topic 2 and 3, although different models select slightly different terms, all these terms can describe the corresponding topic to some extent. For topic 1 (DB), however, the words "accelerator", "performance", and "distributed" of TMSG are more telling than "text" derived by TMBP, and "memory" and "feature" derived by PLSA. Similar subtle differences can be found for the topic 4 as well. Intuitively, TMSG selects more related terms for each topic than other methods, which shows the better performance of TMSG by considering the relationship of entities in the semantic graph.

5 Conclusion

The main contribution of this paper is to show the usefulness of semantic graph for topic modelling. Our proposed TMSG (Topic Model with Semantic Graph) supersedes the existing ones since it takes account both homogeneous networks (i.e., entity to entity relations) and heterogeneous networks (i.e., entity to document relations), and since it exploits both local and global representation of rich knowledge that go beyond networks and spaces.

There are some interesting future work to be continued. First, TMSG only relies on one of the simplest latent topic models (namely PLSA), which makes sense as a first step towards integrating semantic graphs into topic models. In the future, we will study how to integrate the semantic graph into other topic modeling algorithms, such as Latent Dirichlet Allocation. Secondly, it would be also interesting to investigate the performance of our algorithm by varying the weights of different types of entities.

Acknowledgements. We thank the anonymous reviewer for their helpful comments. We acknowledge support from the EPSRC funded project named **A Situation Aware Information Infrastructure Project** (EP/L026015) and the **Integrated Multimedia City Data** (IMCD), a project within the ESRC-funded **Urban Big Data Centre** (ES/L011921/1). This work was also partly supported by NSF grant #61572223. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the view of the sponsor.

References

1. Bao, Y., Collier, N., Datta, A.: A partially supervised cross-collection topic model for cross-domain text classification. In: CIKM 2013, pp. 239–248 (2013)
2. Blei, D.M., Ng, A.Y., Jordan, M.I., Lafferty, J.: Latent dirichlet allocation. **3**, 459–565
3. Cai, L., Zhou, G., Liu, K., Zhao, J.: Large-scale question classification in cqa by leveraging wikipedia semantic knowledge. In: CIKM 2011, pp. 1321–1330 (2011)

4. Chen, X., Zhou, M., Carin, L.: The contextual focused topic model. In: KDD 2012, pp. 96–104 (2012)
5. Deng, H., Han, J., Zhao, B., Yintao, Y., Lin, C.X.: Probabilistic topic models with biased propagation on heterogeneous information networks. In: KDD 2011, pp. 1271–1279 (2011)
6. Guo, W., Diab, M.: Semantic topic models: Combining word distributional statistics and dictionary definitions. In: EMNLP 2011, pp. 552–561 (2011)
7. Hofmann, T.: Unsupervised learning by probabilistic latent semantic analysis. Mach. Learn. **45**, 256–269
8. Hong, L., Dom, B., Gurumurthy, S., Tsioutsiouliklis, K.: A time-dependent topic model for multiple text streams. In: KDD 2011, pp. 832–840 (2011)
9. Hulpus, I., Hayes, C., Karnstedt, M., Greene, D.: Unsupervised graph-based topic labelling using dbpedia. WSDM 2013, pp. 465–474 (2013)
10. Kim, H., Sun, Y., Hockenmaier, J., Han, J.: Etm: Entity topic models for mining documents associated with entities. In: ICDM 2012, pp. 349–358 (2012)
11. Li, F., He, T., Xinhui, T., Xiaohua, H.: Incorporating word correlation into tag-topic model for semantic knowledge acquisition. In: CIKM 2012, pp. 1622–1626 (2012)
12. Li, H., Li, Z., Lee, W.-C., Lee, D.L.: A probabilistic topic-based ranking framework for location-sensitive domain information retrieval. In: SIGIR 2009, pp. 331–338 (2009)
13. Mei, Q., Cai, D., Zhang, D., Zhai, C.: Topic modeling with network regularization. In: WWW 2008, pp. 342–351 (2008)
14. Schuhmacher, M., Ponzetto, S.P.: Knowledge-based graph document modeling. In: WSDM 2014, pp. 543–552 (2014)
15. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Zhong, S.: Arnetminer: extraction and mining of academic social networks. In: KDD 2008, pp. 428–437 (2008)
16. Xing Wei, W., Croft, B.: Lda-based document models for ad-hoc retrieval. In: SIGIR 2006, pp. 326–335 (2009)
17. Wei, X., Liu, X., Gong, Y.: Document clustering based on non-negative matrix factorization. In: SIGIR 2003, pp. 267–273 (2003)