RankTopic: Ranking Based Topic Modeling

Dongsheng Duan†, Yuhua Li*†, Ruixuan Li†, Rui Zhang‡ and Aiming Wen†
†*School of Computer Science and Technology, Huazhong University of Science and Technology, China*
Email: duandongsheng@smail.hust.edu.cn, {idcliyuhua, rxli}@hust.edu.cn, wenaiming@smail.hust.edu.cn
‡*Department of Computing and Information Systems, University of Melbourne, Australia*
Email: rui@csse.unimelb.edu.au

***Abstract*—Topic modeling has become a widely used tool for document management due to its superior performance. However, there are few topic models distinguishing the importance of documents on different topics. In this paper, we investigate how to utilize the importance of documents to improve topic modeling and propose to incorporate link based ranking into topic modeling. Specifically, topical pagerank is used to compute the topic level ranking of documents, which indicates the importance of documents on different topics. By retreating the topical ranking of a document as the probability of the document involved in corresponding topic, a generalized relation is built between ranking and topic modeling. Based on the relation, a ranking based topic model *RankTopic* is proposed. With RankTopic, a mutual enhancement framework is established between ranking and topic modeling. Extensive experiments on paper citation data and Twitter data are conducted to compare the performance of RankTopic with that of some state-of-the-art topic models. Experimental results show that RankTopic performs much better than some baseline models and is comparable with the state-of-the-art link combined relational topic model (RTM) in generalization performance, document clustering and classification by setting a proper balancing parameter. It is also demonstrated in both quantitative and qualitative ways that topics detected by RankTopic are more interpretable than those detected by some baseline models and still competitive with RTM.**

***Keywords*-Ranking; Topic Modeling; Document Network; Clustering; Classification**

I. Introduction

Document network is defined as a collection of documents that are connected by links. Document networks become ubiquitous nowadays due to the widespread use of online databases, such as academic search engines [1]. In general, documents can have various kinds of textual contents, such as research papers, web pages or tweets. Documents can also be connected via a variety of links. For example, papers can be connected together via citations, web pages can be linked by hyper-links, and tweets can link to one another according to the retweet relationship.

Given a document network, topic modeling aims at discovering semantically coherent clusters of correlated words known as topics. Traditional topic models include PLSA (Probabilistic Latent Semantic Analysis) [2] and LDA (Latent Dirichlet Allocation) [3]. By using topic modeling, documents can be modeled as a distribution over topics instead of that over words. As features of documents, topics are usually much lower in dimension and much more interpretable than words.

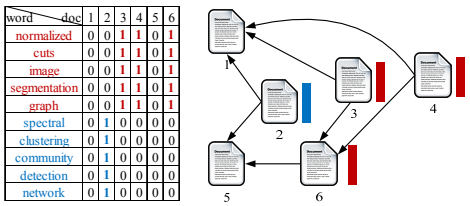
However, most topic models treat documents as equally important, while in practical situations documents have different degrees of importance on different topics, thus treating them as equally important may inherently hurt the performance of topic modeling. To quantify the importance of documents on different topics, topical pagerank [4] can be used, which is an extension of a well known ranking algorithm pagerank [5]. Although pagerank is initially proposed for the purpose of ranking web pages, it can be also used to rank research publications since concepts and entities in both domains are similar [6]. In this work, we propose to incorporate link based ranking into topic modeling.

Specifically, topical pagerank is employed to compute the importance scores of documents over topics, which are then leveraged to guide the topic modeling process. The proposed topic model is called *ranking based topic model*, denoted as *RankTopic* for short. Compared to existing topic models, RankTopic distinguishes the importance of documents while performing topic modeling. The philosophy behind the methodology is that the higher ranked documents are given more weights than the lower ranked ones.

As a motivating example, let's see a small artificial network with six documents as Figure 1 shows. The left side of the figure is the *word-document matrix*, and the right side is a fictional link structure among those imaginary documents. Traditional topic model (i.e. PLSA or LDA) discovers two topics, which are colored by red and blue respectively. The two topics can be interpreted as "image segmentation" and "community detection" from corresponding words in them. The colored bars beside documents indicate their topic proportions. Since documents 1 and 5 have no words, both of them are not labeled by any topics.

However, from the link structure, we have reason to believe that documents 1 and 5 should have been labeled by some topics because they are cited by documents with the two topics. As a link combined topic model, iTopic [7] can alleviate this issue to some degree. Figure 2(a) illustrates the topic detection result of iTopic, from which we can see that documents 1 and 5 are labeled by the two topics but with

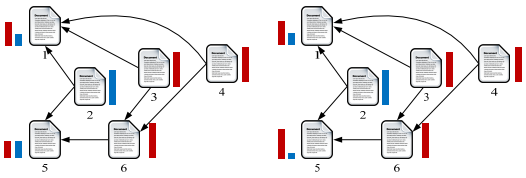
*Corresponding Author



word \ doc	1	2	3	4	5	6
normalized	0	0	1	1	0	1
cuts	0	0	1	1	0	1
image	0	0	1	1	0	1
segmentation	0	0	1	1	0	1
graph	0	0	1	1	0	1
spectral	0	1	0	0	0	0
clustering	0	1	0	0	0	0
community	0	1	0	0	0	0
detection	0	1	0	0	0	0
network	0	1	0	0	0	0

(a) Documents (b) Link structure

Figure 1. An artificial document network. There are two topics in these documents, which are colored by red and blue respectively. Documents are labeled by corresponding colored bars beside them.



(a) iTopic (b) RankTopic

Figure 2. The topic proportions of documents output by iTopic and RankTopic respectively. Higher bars indicate more proportions.

different proportions. Document 1 has more proportions on red topic than on blue one while document 5 has the same proportion on them. Notice that document 1 is cited by two red topics (documents 3 and 4) and one blue (document 2), while document 5 is cited by one red (document 6) and one blue (document 2). iTopic treats neighboring documents as equally important such that the topic proportions of both documents 1 and 5 are computed as averages of topic proportions of their neighbors.

However, documents can have various importance on different topics, so treating them as equally important may obtain inaccurate topics. RankTopic incorporates the ranking into topic modeling such that it can well distinguishes the importance of documents. Figure 2(b) shows the topic detection result of RankTopic, from which we can see that document 5 has much more proportions on red topic than blue one. The underlying reason is that document 6 ranks high on red topic as it is cited by two red topics, while document 2 ranks low because it is not cited by any documents. There are more evidence showing that document 5 is more likely about red topic than blue one. In the aspect of capturing such evidence, RankTopic performs reasonably better than iTopic and other network regularization based topic models, such as NetPLSA [8], which motivates our study on RankTopic.

In the above example, we clearly see that RankTopic can well incorporate the importance of documents into topic modeling and addresses the drawbacks of some existing topic models. We also experimentally demonstrate that RankTopic outperforms some baseline topic models and is comparable with one of the sate-of-the-art link combined topic model RTM (Relational Topic Model) [9] in generalization performance, document clustering and classification, and topic interpretability, which are all validated either quantitatively or qualitatively in the experimental section. Compared with existing topic models, RankTopic has the following distinguished characteristics.

- Existing topic models assume that documents plays equally important role in topic modeling. In contrast, RankTopic incorporates the ranking of documents into topic modeling and benefit from such combination.
- Previous works treat topic modeling and ranking as two independent issues while RankTopic puts them together and makes them mutually enhanced in a unified framework.
- RankTopic is flexible since ranking and topic modeling are orthogonal to each other such that different ranking and topic modeling methods can be used according to specific application requirements.

The rest of the paper is organized as follows. Section II reviews the related works. Section III presents the preliminaries about topic modeling and ranking. We propose RankTopic model and present parameter learning algorithm for the proposed model in Section IV. Experimental settings and results are demonstrated in Section V and we conclude this paper in Section VI.

II. Related Works

Topic models have been widely studied in the text mining community due to its solid theoretical foundation and promising performance. PLSA [2] and LDA [3] are two well known basic topic models. Since they are proposed, various kinds of extensions have been proposed by incorporating more contextual information, such as time [10], [11], [12], [13], authorship [14], and links [7], [15], [8], [16], [9]. [17] combines collaborative filtering and LDA for recommending scientific publications. The present work also incorporates links into topic modeling but uses different way from previous works. Although most earlier link combined topic models can capture the topical correlations between linked documents, there are few works leveraging the topical ranking of documents to guide the topic modeling process. The most similar work to ours may be the TopicFlow model [18]. The distinguished features of present work from TopicFlow lie in the following folds. First, RankTopic provides a more flexible combination between ranking and topic modeling while TopicFlow couples flow network and topic modeling tightly. This feature makes RankTopic more extendable. Second, RankTopic builds a generalized relation between ranking and topic modeling rather than a hard relation like TopicFlow. Third, the topic specific influence of documents computed by TopicFlow can actually serve as the topical ranking in RankTopic as an alternative of topical pagerank adopted by us.

Our work is also tightly related to ranking technology. The most well known link based ranking algorithms are PageRank [5] and HITS [19]. Both algorithms are based on

the phenomenon that rich gets richer. Topical ranking [4] extends the algorithms by calculating a vector of scores to distinguish the importance of documents on different topics. [20] proposes random walk with topic nodes and random walk at topical level to further rank documents over heterogenous network. RankClus [21], [22] further extends the method to heterogenous information networks to rank one kind of node with respect to another. Compared to RankClus which performs ranking based on hard clustering, we incorporate ranking into topic modeling which is a soft clustering. Another difference is that RankClus is a clustering algorithm based on only links while RankTopic is a topic modeling algorithm based on both links and texts.

We would like to mention PCL-DC [23], which is a community detection model by combining links and textual contents. The node popularity introduced in PCL-DC can also be regarded as link based ranking. However, PCL-DC introduces the popularity variable in the link based community detection model (PCL) but does not directly use it in the discriminative content (DC) model, while RankTopic explicitly incorporates ranking into the generative model of textual contents. Another difference is that PCL-DC is a discriminative model while RankTopic is a generative one such that PCL-DC can not generalize to unknown data.

III. Preliminaries

A. Topic Modeling

Topic modeling aims at extracting conceptually coherent topics shared by a set of documents. In the following, we describe topic model PLSA [2] upon which RankTopic is built. We choose the most basic topic model PLSA rather than the extended one, such as LDA, because we would like to eliminate the effect of other factors, such as the Dirichlet prior in LDA.

Given a collection of M documents D, let V denote the total number of unique words in the vocabulary and K represent the number of topics, the goal of PLSA is to maximize the likelihood of the collection of documents with respect to model parameters Θ and B.

$$P(D|\Theta, B) = \prod_{i=1}^{M}\prod_{w=1}^{V}\left(\sum_{z=1}^{K}\theta_{iz}\beta_{zw}\right)^{s_{iw}} \quad (1)$$

where $\Theta = \{\theta\}_{M\times K}$ is the topic distribution of documents, $B = \{\beta\}_{K\times V}$ is the word distribution of topics, and s_{iw} represents the times that word w occurs in document i.

After the inference of PLSA, each topic is represented as a distribution over words in which top probability words form a semantically coherent concept, and each document can be represented as a distribution over the discovered topics.

B. Ranking

Pagerank [5] is a well known link based ranking algorithm. The main idea of pagerank is that, the importance score of a document equals the sum of those propagated from its in-link neighbors. However, ranking documents by a single global importance score may not make much sense because documents should be ranked sensitive to their contents. Based on this consideration, topical pagerank [4] is proposed.

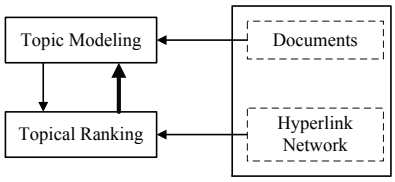


Figure 3. Mutual enhancement framework for ranking and topic modeling

As the input of topical pagerank, each document i is associated with a topic distribution θ_i, which can be obtained via topic modeling methods. Taking the topic distribution of documents into account, topical pagerank produces a ranking vector for each document, in which each element represents the importance score of the document on each topic. Letting γ_{zi} denote the ranking of document i on topic z, topical pagerank is formally expressed as

$$\gamma_{zi}^{(t)} = \lambda\sum_{j\in I_i}\frac{\alpha\gamma_{zj}^{(t-1)} + (1-\alpha)\theta_{jz}\gamma_{\cdot j}^{(t-1)}}{|O_j|} + (1-\lambda)\frac{\theta_{iz}}{M} \quad (2)$$

where α and λ are parameters that control the process of prorogating the ranking score, which are both empirically set to 0.85. $\gamma_{\cdot j} = \sum_{z=1}^{K}\gamma_{zj}$ denotes the global ranking of document j, I_i is the set of in-link neighbors of document i, $|O_j|$ denotes the number of out-link neighbors of document j, and θ_{jz} is the topic proportion of document j on topic z and M is the total number of documents.

The process of topical pagerank is illustrated in Figure 3 excluding the thick line. It can be seen that topical pagerank first performs topic modeling and then performs ranking in topic level, thus it regards ranking and topic modeling separately. It is worthy to point out that the original topical pagerank [4] method uses supervised learning method based on predefined categories from Open Directory Project (ODP) other than topic modeling methods to obtain the topic distribution of documents.

IV. Ranking based Topic Modeling

In this section, we incorporate the ranking into topic modeling and elaborate the proposed ranking based topic model. Specifically, we first build a relation between ranking and topic modeling, based on which ranking based topic model RankTopic is then presented and derived in detail, following which the learning algorithm of RankTopic is presented.

A. Relation of Ranking and Topic Modeling

To incorporate the ranking into topic modeling, it is essential to build the relation between them. However, there

is no closed solution for establishing this relation. Here, we present a natural way to achieve this end.

Notice that the ranking γ_{zi} can be interpreted as the probability $P(i|z)$ of the node i involved in the topic z by normalizing the ranking such that $\sum_{i=1}^{M} P(i|z) = 1, \forall z$. By using the sum and product rules of the Bayesian theorem, the topic proportion $P(z|i)$ can be expressed in terms of γ_{zi}.

$$\theta_{iz} = P(z|i) = \frac{P(i|z)p(z)}{\sum_{i'=1}^{M} P(i|z)p(z)} = \frac{\gamma_{zi}\pi_z}{\sum_{z'=1}^{K} \gamma_{z'i}\pi_{z'}} \quad (3)$$

where $\pi_z = P(z)$ is the prior probability of topic z.

By using the above interpretation, the topic proportion of a document is decomposed into the multiplication of topical ranking and the prior distribution of topics. However, there is still a problem for the above equation. Topical ranking is computed based on the link structure of the document network, which inevitably have noise in practical situations. We observe some self-references in the ACM digital library, which is usually caused by some error editing behavior. Inappropriate and incomplete references may also exist. Therefore, equating between the topical ranking γ_{zi} and the conditional probability $P(i|z)$ also bring much noise into the topic modeling. One possible solution for this problem is to detect the noise links and remove them from the document network. However, spam detection itself is a challenging issue, which is out of the scope of this paper.

To reduce the effects of noise, we model the degree of our belief on the ranking instead of removing the noise links. Specifically, we transform Equation 3 to a more generalized one by introducing a parameter ξ ranging from 0 to 1 to indicate our belief on the ranking as follows.

$$\theta_{iz} = P(z|i) \propto \xi\gamma_{zi}\pi_z + (1-\xi)p(i|z)\pi_z = [\xi\gamma_{zi} + (1-\xi)\phi_{zi}]\pi_z \quad (4)$$

where $\phi_{zi} = p(i|z)$ has the same interpretation as γ_{zi}, but it is a hidden variable rather than an observed one.

In Equation 4, if $\xi = 0$, the topic proportions are the same as that in PLSA, and if $\xi = 1$, the topic proportions are completely dependent on the topical ranking. Intermediate values of ξ balance between the above two extreme cases. The larger the value of ξ, the more information of ranking is incorporated into the topic modeling. Therefore, Equation 3 is actually a special case of Equation 4 by setting ξ to 1.

B. RankTopic Model

Based on the generalized relation between ranking and topic proportion, we can replace θ in PLSA with the right side of Equation 4, which results in the ranking based topic model *RankTopic*. Figure 4 shows the graphical representation of the RankTopic model. Different from the traditional topic models, the probability $p(i|z)$ of a document i involved in a topic z is governed by the weighted mixture of topical ranking γ_{zi}, and the hidden variable ϕ_{zi} in the RankTopic model such that the effects of ranking on topic modeling is integrated.

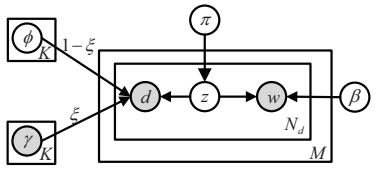


Figure 4. Ranking based topic model

In this model, the topical ranking γ of documents is labeled as observational variable (shaded in Figure 4) since it can be obtained by the topical pagerank algorithm introduced in Section III-B, although in an overall view topical ranking is in fact unknown. By incorporating topical ranking γ_{zi} into the topic modeling, the link information is naturally taken into account since the topical ranking process is performed on the link structure.

For RankTopic model, the likelihood of a collection of documents D with respect to the model parameters is

$$P(D|\gamma, \pi, \phi, \beta) = \prod_{i=1}^{M}\prod_{w=1}^{V}\left(\sum_{z=1}^{K}[\xi\gamma_{zi} + (1-\xi)\phi_{zi}]\,\pi_z\beta_{zw}\right)^{s_{iw}} \quad (5)$$

where the definition of all the notations can be found in the previous parts of this paper. Next, the maximum likelihood estimation is adopted to derive the model parameters involved in RankTopic.

C. Derivation of RankTopic

To obtain the (local) maximum of the likelihood in Equation 5, the expectation maximization (EM) based algorithm is employed. Detailed derivation of the EM updating rules is as follows.

The logarithm of the likelihood function is

$$\begin{aligned} L &= \log P(D|\gamma, \pi, \phi, \beta) \\ &= \sum_{i=1}^{M}\sum_{w=1}^{V} s_{iw} \log \sum_{z=1}^{K} \beta_{zw}\,[\xi\gamma_{zi} + (1-\xi)\phi_{zi}]\,\pi_z \end{aligned} \quad (6)$$

In the E-step, the posterior distribution $P(z|i, w)$ of topics conditioned on each document-word pair (i, w) is computed by Equation 7.

$$\psi_{iwz}^{(t)} = P^{(t)}(z|i, w) \propto \beta_{zw}^{(t)}\left[\xi\gamma_{zi} + (1-\xi)\phi_{zi}^{(t)}\right]\pi_z^{(t)} \quad (7)$$

Then, the lower bound of L can be derived by using Jensen inequality twice as following,

$$\begin{aligned} L &= \sum_{i=1}^{M}\sum_{w=1}^{V} s_{iw} \log \sum_{z=1}^{K} \psi_{iwz}^{(t)} \frac{\beta_{zw}\,[\xi\gamma_{zi} + (1-\xi)\phi_{zi}]\,\pi_z}{\psi_{iwz}^{(t)}} \\ &\geq \sum_{i=1}^{M}\sum_{w=1}^{V} s_{iw} \sum_{z=1}^{K} \psi_{iwz}^{(t)} \log \beta_{zw}\,[\xi\gamma_{zi} + (1-\xi)\phi_{zi}]\,\pi_z \\ &\quad - \sum_{i=1}^{M}\sum_{w=1}^{V} s_{iw} \sum_{z=1}^{K} \psi_{iwz}^{(t)} \log \psi_{iwz}^{(t)} \\ &\geq \sum_{i=1}^{M}\sum_{w=1}^{V} s_{iw} \sum_{z=1}^{K} [\xi\psi_{iwz}^{(t)} \log \beta_{zw}\gamma_{zi}\pi_z \\ &\quad + (1-\xi)\psi_{iwz}^{(t)} \log \beta_{zw}\phi_{zi}\pi_z] - \sum_{i=1}^{M}\sum_{w=1}^{V} s_{iw} \sum_{z=1}^{K} \psi_{iwz}^{(t)} \log \psi_{iwz}^{(t)} \end{aligned}$$

In the M-step, the lower bound of L is maximized under the constraints $\sum_{w=1}^{V} \beta_{zw} = 1$, $\sum_{z=1}^{K} \pi_z = 1$ and $\sum_{i=1}^{M} \phi_{zi} = 1$. Through introducing Lagrange multipliers, the constrained maximization problem is converted to the following one.

$$\max_{\theta,\pi} \sum_{i=1}^{M} \sum_{w=1}^{V} s_{iw} \sum_{z=1}^{K} \left[\xi \psi_{iwz}^{(t)} \log \beta_{zw} \gamma_{zi} \pi_z + (1-\xi) \psi_{iwz}^{(t)} \log \beta_{zw} \phi_{zi} \pi_z \right] + \sum_{z=1}^{K} \lambda_z \left(\sum_{w=1}^{V} \beta_{zw} - 1 \right) + \lambda \left(\sum_{z=1}^{K} \pi_z - 1 \right) + \sum_{z=1}^{K} \lambda'_z \left(\sum_{i=1}^{M} \phi_{zi} - 1 \right)$$

The above maximization problem has a closed form solution as follows, which gives out the update rules that monotonically increase L.

$$\beta_{zw}^{(t+1)} \propto \sum\nolimits_{i=1}^{M} s_{iw} \psi_{iwz}^{(t)} \tag{8}$$

$$\pi_{z}^{(t+1)} \propto \sum\nolimits_{i=1}^{M} \sum\nolimits_{w=1}^{V} s_{iw} \psi_{iwz}^{(t)} \tag{9}$$

$$\phi_{zi}^{(t+1)} \propto \sum\nolimits_{w=1}^{V} s_{iw} \psi_{iwz}^{(t)} \tag{10}$$

As the parameter updating process converges, the topic proportion θ can be computed by using Equation 4.

D. The Learning Algorithm of RankTopic

With RankTopic, we can build a mutual enhancement framework by organizing topic modeling and ranking into an alternative process illustrated in Figure 3. By introducing RankTopic as the thick line shows, the sequential framework from topic modeling to ranking is transformed to a mutual enhancement framework.

From the implementation view, we provide the matrix form of the parameter estimation equations. The parameters involved in the overall framework include topic-word distributions $\mathrm{B} = \{\beta\}_{K \times V}$, hidden variable $\Phi = \{\phi\}_{K \times N}$, topic prior distributions $\Pi = \{\pi\}_K$, and topical ranking $\Gamma = \{\gamma\}_{K \times N}$. Let $\mathrm{S} = \{s\}_{N \times V}$ denote the document-word matrix in which s_{iw} represents the time word w occurs in document i. Let $\mathrm{L} = \{l\}_{N \times N}$ denote the link structure among those documents in which $l_{ij} = 1$ represents that there is a link from document i to document j and $l_{ij} = 0$ represents there is not.

It can be proved that Equation 8, 9, 10 and 2 have the following four matrix forms respectively.

$$\mathrm{B} = \mathrm{B}. * \left(\mathrm{Y} (\mathrm{S}. / (\mathrm{Y}^{\mathrm{T}} \mathrm{B})) \right) \tag{11}$$

where $\mathrm{Y} = (\xi\Gamma + (1-\xi)\Phi) \left[\begin{array}{ccc} \Pi & \cdots & \Pi \end{array} \right]$, and $.*$ and $./$ represent element wise multiplication and division operation between two matrices respectively.

$$\Pi = \mathrm{diag}\{\mathrm{Y}(\mathrm{S}./(\mathrm{Y}^{\mathrm{T}}\mathrm{B}))\mathrm{B}^{\mathrm{T}}\} \tag{12}$$

where $\mathrm{diag}\{\cdot\}$ returns the main diagonal of a matrix.

$$\Phi = \mathrm{Y}. * \left(\mathrm{B} \left(\mathrm{S}./(\mathrm{Y}^{\mathrm{T}}\mathrm{B}) \right)^{\mathrm{T}} \right) \tag{13}$$

Algorithm 1: The learning algorithm of RankTopic

Input: A document network L with M documents including totally V unique words, and the expected number K of topics and parameter ξ.
Output: Topic-word distributions B, Document-topic distributions Θ.
initialization: Perform PLSA to obtain B and Θ;
repeat
  repeat
    $\Gamma = \lambda \left(\alpha\Gamma + (1-\alpha)\mathrm{X}\right) \widehat{\mathrm{L}} + \frac{1-\lambda}{M} \Theta^{\mathrm{T}}$;
    Normalize Γ such that $\forall z, i, \sum_{z=1}^{K} \sum_{i=1}^{N} \gamma_{zi} = 1$;
  until *Satisfying condition 1*;
  repeat
    $\mathrm{B} = \mathrm{B}. * \left(\mathrm{Y} (\mathrm{S}. / (\mathrm{Y}^{\mathrm{T}} \mathrm{B})) \right)$;
    Normalize B such that $\forall z, \sum_{w=1}^{V} \beta_{zw} = 1$;
    $\Pi = \mathrm{diag}\{\mathrm{Y}(\mathrm{S}./(\mathrm{Y}^{\mathrm{T}}\mathrm{B}))\mathrm{B}^{\mathrm{T}}\}$;
    Normalize Π such that $\sum_{z=1}^{K} \pi_z = 1$;
    $\Phi = \mathrm{Y}. * \left(\mathrm{B} \left(\mathrm{S}./(\mathrm{Y}^{\mathrm{T}}\mathrm{B}) \right)^{\mathrm{T}} \right)$;
    Normalize Φ such that $\forall z, \sum_{i=1}^{N} \phi_{zi} = 1$;
  until *Satisfying condition 2*;
  $\Theta = (\xi\Gamma + (1-\xi)\Phi) \left[\begin{array}{ccc} \Pi & \cdots & \Pi \end{array} \right]$;
until *Satisfying condition 3*;
return B Θ;

$$\Gamma = \lambda \left(\alpha\Gamma + (1-\alpha)\mathrm{X}\right) \widehat{\mathrm{L}} + \frac{1-\lambda}{M} \Theta^{\mathrm{T}} \tag{14}$$

where $\mathrm{X} = \left(\left[\begin{array}{ccc} \mathrm{sum}(\Gamma) & \cdots & \mathrm{sum}(\Gamma) \end{array} \right] \Theta \right)^{\mathrm{T}}$, $\mathrm{sum}(\cdot)$ returns sums along the columns of a matrix, and $\widehat{\mathrm{L}}$ is the row normalization matrix of link structure L.

According to the mutual enhancement framework and matrix forms of the updating rules presented above, the learning algorithm of RankTopic is summarized in Algorithm 1. In the following, we present the three termination conditions in the algorithm.

Condition 1: This condition is to test whether the topical ranking Γ converges. We compute the differences between the topical ranking of the current iteration and the previous one, and sum these differences over all the cells. If the difference is lower than a predefined small value (1e-2 in our experiments), this condition is satisfied.

Condition 2: This condition is to test whether the ranking based topic modeling process converges. For each iteration, we compute the log-likelihood of the observed documents with respect to the current parameters B, Γ, Φ and Π via Equation 6, and then compute the relative change of the log-likelihood between two continuous iterations as the fraction of the difference between the two log-likelihoods to the average value of them. If the relative change is lower than a predefined small value (1e-4 in our experiments), this condition is satisfied.

Condition 3: This condition is to test whether the whole process archives a (local) optimal solution. For each iteration, we compute the log-likelihood of the ranking-integrated document matrix with respect to the current parameters B and Θ. The ranking-integrated document-word matrix is computed by using topical pagerank on the link structure

and original document-word matrix. The ranking-integrated document-word matrix is actually an imaginary document-word matrix which contains the observational information from both documents and links. The higher the computed log-likelihood, the better the current solution. If the incremental quantity of the log-likelihood is lower than a predefined threshold (1e-3 in our experiments), this condition is satisfied.

From a brief analysis, the time complexity of the algorithm turns out to be $O(E + N \times V)$, which is linear in the total number of links and words in the observed document network.

V. Experiments

In this section, we conduct experimental studies of RankTopic in various aspects, and compare it with some state-of-the-art topic models, namely PLSA, LDA, iTopic and RTM (Relational Topic Model) [9]. The code for both PLSA and LDA is downloaded from http://lear.inrialpes.fr/~verbeek/software.php while that for iTopic is provided by its author. All the models except RTM are implemented in Matlab, while the R package for RTM implemented by its author is used in the following experiments (http://cran.r-project.org/web/packages/lda/). It is worthy to point out that all the models except RTM are learned by EM (Expectation Maximization) based algorithm while RTM is learned by Gibbs sampling method because the R package for RTM provides a fast collapsed Gibbs sampler in C language. In the experiments, we use two genres of data sets, i.e. three public paper citation data sets and one twitter data set.

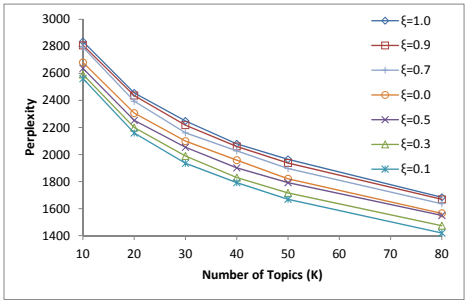
ArnetMiner: This is a subset of the Citation-network V1 (http://www.arnetminer.org/citation) released by ArnetMiner [1]. After some preprocessing, there are 6,562 papers and 8,331 citations left and 8,815 unique words in the vocabulary.

Citeseer: This data set consists of 3,312 scientific publications and 4,715 links. The dictionary consists of 3,703 unique words. These publications have been categorized into 7 classes according to their research directions in advance.

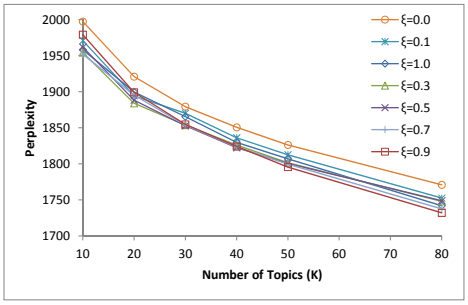
Cora: There are 2,708 papers, and 5,429 citations in this subset of publications. The dictionary consists of 1433 unique words. These publications have been labeled as one of 6 categories in advance.

Twitter: The twitter data we used is released by [24], which can be downloaded from http://arnetminer.org/heterinf. In this data set, users associated with their published tweets are regarded as documents and the '@' relationship among users as links. After some preprocessing like stop word removing, we obtain 814 users in total and 5,316 unique words in the vocabulary. There are 4,206 '@' relationships between those users.

Both Citeseer and Cora data sets used in our experiments is the same as that used in [23].



(a) Parameter Study on ArnetMiner



(b) Parameter Study on Twitter

Figure 5. Perplexity results of RankTopic on ArnetMiner and Twitter data sets by setting some typical values of parameter ξ and K (number of topics). All these results are average values computed under 10-fold cross validation.

A. Generalization Performance

In this subsection, we compare RankTopic with several baseline topic models in terms of generalization performance. Perplexity [25] is a widely used measure for evaluating the generalization performance of a probabilistic model. Lower perplexity indicates better generalization performance.

Given a trained topic model $\{\Theta, B\}$, the likelihood $P(d_i^{test}|\Theta, B)$ of document d_i^{test} in the test corpus is computed as follows.

$$P(d_i^{test}|\Theta, B) = \prod_{w=1}^{V} \left(\sum_{z=1}^{K} \theta_{iz}\beta_{zw} \right)^{s_{iw}^{test}}$$

where s_{iw}^{test} represents the times that word w occurs in the i-th testing document. Perplexity is formally defined as as follows.

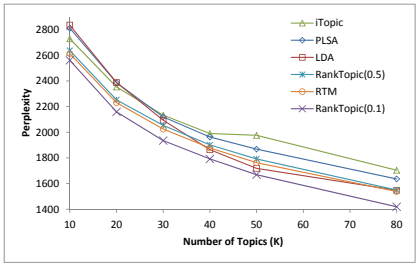
$$\text{Perplexity} = \exp\{-\frac{\sum_{i=1}^{M^{test}} \log(P(d_i^{test}|\Theta, B))}{\sum_{i=1}^{M^{test}} N_i^{test}}\}$$

where M^{test} is the number of documents in the test corpus and N_i^{test} is the number of words in d_i^{test}.

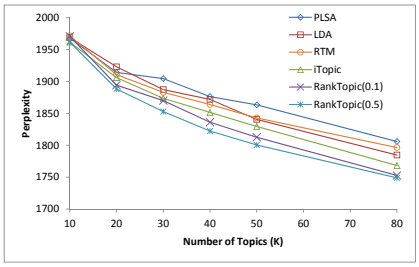
In our experiments, we perform 10-fold cross validation. Before comparing RankTopic with other topic models, we first study how the value of parameter ξ affects the generalization performance of RankTopic. Figure 5 shows parameter study results for some typical values of ξ on ArnetMiner and Twitter data. From the results, we observe the following phenomenons.

Both the results on ArnetMiner and Twitter data sets consistently show that RankTopic could obtain lower perplexity than the special case when ξ equals 0.0, which actually degenerates to PLSA but with additional termination condition for outside loop (see condition 3 in section IV-D). These results show that link based ranking can indeed be used to improve the generalization performance of basic topic models. However, we also observe different effects of ξ on RankTopic's generalization performance for different data sets. For ArnetMiner data, the lower the value of ξ, the better RankTopic's generalization performance except for $\xi = 0.0$. For Twitter data, the best generalization performance is obtained when $\xi = 0.9$ and perplexity is less sensitive to ξ except for the special case of $\xi = 0.0$. Whether RankTopic is sensitive to ξ may significantly depends on the consistency between links and texts and the noises in them. Nevertheless, we provide a tuning way for adapting RankTopic into practical senecios.

Figure 6 illustrates the perplexity results of the compared topic models. Results show that RankTopic with appropriately set ξ performs best among all the compared models, which indicates its superior generalization performance over the baseline topic models. The underlying reasons for the results are analyzed as follows. By introducing Dirichlet prior, LDA performs better than PLSA when K value increases. However, the prior adopted by LDA is non-informative. RankTopic can also be regarded as incorporating prior into PLSA, but topical ranking is more informative than Dirichlet prior. Both RTM and iTopic incorporate link structure into topic modeling. However, iTopic assumes that the neighbors of a node play equally important role in affecting the topics of that node, which is usually not the truth in practical document networks. The topics detected by RTM are governed by both link regression process and the document contents, but RTM does not model the weights of the two parts such that its generalization performance depends on the accuracy of links and contents. In contrast, RankTopic provides a turning weight of the incorporation of ranking such that it is more flexible than RTM.



(a) Perplexity Comparison on ArnetMiner



(b) Perplexity Comparison on Twitter

Figure 6. Perplexity results of RankTopic and some baseline topic models on ArnetMiner and Twitter data sets by setting various numbers of topics (K). All these results are averages computed under 10-fold cross validation. For RankTopic, the results of $\xi = 0.1$ and $\xi = 0.5$ are shown for comparison purpose.

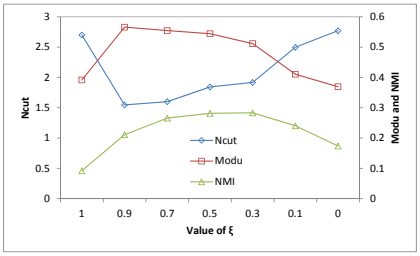


Figure 7. Clustering performance of RankTopic with some typical values of ξ on Citeseer data. For Ncut, the lower the better. For both Modu and NMI, the higher the better. Notice that the left Y-axis is just for Ncut, while the right one is for both Modu and NMI.

B. Document Clustering

Besides the generalization performance, topic models can also be evaluated by using their application performance. The most widely used applications of topic models include document clustering and classification. In this and subsequent subsection, we study the performance of RankTopic on document clustering and classification respectively.

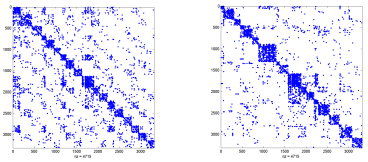
By using topic models, documents can be represented as topic proportion vectors, upon which document clustering can be performed. Specifically, we adopt k-means as the clustering algorithm. For a network, normalized cut (Ncut) [26], modularity (Modu) [27], are two well known measures for evaluating the clustering results. Lower normalized cut and higher modularity indicates better clustering result. When the background label information is known for documents, normalized mutual information (NMI) [28] can also be used to evaluate the clustering result. The higher the NMI, the better the clustering quality. In these experiments, the number of clusters and topics are set to 6 for the Citeseer data, 7 for the Cora data, which are suggested by [23], and 10 for twitter data. Since there are no significant clusters in ArnetMiner data, clustering results on ArnetMiner is not shown.

We first study the effect of parameter ξ. Figure 7 shows the results. For both Ncut and Modu, RankTopic with $\xi = 0.9$ performs best on Citeseer data. For NMI, RankTopic with $\xi = 0.3$ performs best on Citeseer data. Overall, RankTopic with $\xi = 0.5$ compromises among the three evaluation measures. We obtain similar results on Cora and Twitter data.

We then compare the clustering performance of Rank-

Table I
CLUSTERING PERFORMANCE OF DIFFERENT MODELS ON CITESEER AND CORA DATA SETS. FOR NCUT, THE LOWER THE BETTER. FOR BOTH MODU AND NMI, THE HIGHER THE BETTER. FOR RANKTOPIC, $\xi = 0.5$. TR REPRESENTS THE TOPICAL RANKING MODEL.

Models	Citeseer			Cora			Twitter	
	Ncut	Modu	NMI	Ncut	Modu	NMI	Ncut	Modu
PLSA	2.92	0.35	0.14	4.85	0.16	0.11	5.88	0.35
LDA	2.68	0.38	0.21	4.30	0.24	0.19	4.76	0.37
iTopic	2.09	0.48	0.26	4.01	0.29	0.21	4.60	0.45
TR	1.99	0.50	0.17	4.74	0.18	0.14	5.37	0.38
RTM	1.63	0.54	**0.31**	**2.98**	0.47	**0.32**	4.24	0.47
RankTopic	**1.60**	**0.55**	0.28	3.01	**0.47**	0.30	**2.77**	**0.53**



(a) iTopic (b) RankTopic

Figure 8. Clustering results of iTopic and RankTopic with $\xi = 0.5$ on Citeseer data. The more a matrix looks like a block diagonal matrix, the better the clustering result summarizes the links.

Topic with the baseline models. Table I reports our experimental results. For the purpose of comparison, results of RankTopic with $\xi = 0.5$ are selected to be shown. From the results, we can see that RankTopic performs better than PLSA, LDA, iTopic, topical ranking (TR) and is comparable with RTM. More importantly, RankTopic outperforms both of its ingredients, i.e. PLSA and topical ranking, which indicates that combining PLSA and ranking has much better clustering performance than each of them. Overall, the link combined topic models have better clustering performance than link ignored ones. NMI is not shown for Twitter since there is no background labels for users in that data.

We finally study the clustering results qualitatively in a visualized way. Since link structure can reflect the clusters of documents to some degree, the adjacency matrix of document network is taken for visualization. For example, clustering results of iTopic and RankTopic on Citeseer data are illustrated in Figure 8. Again, the clustering result for RankTopic with $\xi = 0.5$ is only shown for comparison. The documents clustered in the same class are arranged to be adjacent to each other in the visualized matrixes. The more a matrix looks like a block diagonal matrix, the better the clustering result summarizes the link structure. The results of PLSA and LDA look even worse than that for iTopic and that of RTM looks more or less the same as RankTopic. The visualization results are consistent with the quantitative results in Table I.

However, there are large volume of community detection algorithms, such as spectral clustering [29] and PCL-DC [23], which aims at partitioning a network into clusters according to the links only. We do not compare RankTopic with them because the community detection algorithms directly perform clustering on links by optimizing measures like normalized cut and modularity. One drawback of those

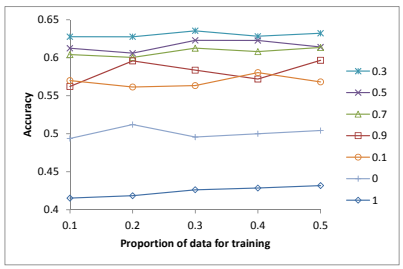


Figure 9. Classification accuracy of RankTopic with some typical values of ξ on Citeseer data set for different proportions of training data. Accuracy is defined as the fraction of the correctly classified documents to the total number of documents. The higher the accuracy, the better the classification quality.

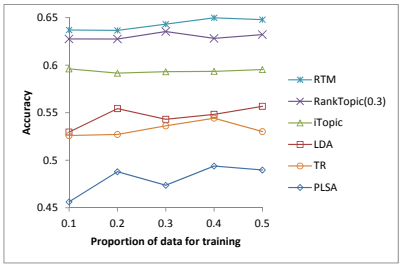
community detection algorithms is that they can only describe the community structure of the observational data but can not generalize the results to unseen data, which actually can be done by topic modeling methods. In this sense, it is not fair to compare topic modeling methods with the community detection algorithms.

C. Document Classification

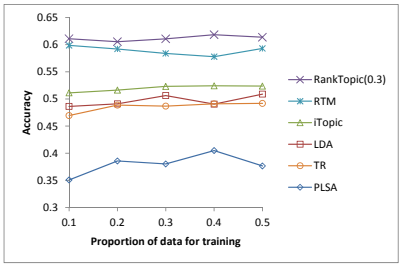
In the previous subsection, we have validated that topics detected by RankTopic serves as better features for document clustering than some baseline topic models. In this subsection, we further study the performance of RankTopic on document classification. We use an open source package, MATLABArsenal (http://finalfantasyxi.inf.cs.cmu.edu/), to conduct the following experiments. Specifically, we select SVM_LIGHT with RBF kernel as the classification method, and set kernel parameter as 0.01 and cost factor as 3. Recall that label information for publications in Citeseer and Cora data sets are known in advance, it is natural to choose the two data sets for classification purpose.

Similarly, we first study the effect of parameter ξ by empirically setting them to some typical values. Figure 9 shows the results on Citeseer Data. From the results, we see that RankTopic with $\xi = 0.3$ perform best in terms of classification accuracy. Overall, RankTopic performs well when ξ is at the middle of the range [0,1] and performs bad when ξ is close or equal to either 0 or 1. We obtain similar results on Cora data. Based on the results, we also compare RankTopic with the baseline models. Figure 10 shows the comparison results. It can be seen that the classification results built on topic features extracted by RankTopic are better than all the baseline topic models except RTM on Citeseer data set. Similar with the clustering results, the classification performance of RankTopic is comparable with RTM, which is one of competitive link combined topic models.

From both the document classification and document clustering results, we conclude that topics detected by RankTopic indeed serve as better features for documents than those detected by some baseline topic models, while



(a) Classification on Citeseer



(b) Classification on Cora

Figure 10. Classification accuracy of different topic models on Citeseer and Cora data sets for different proportions of training data. The higher the accuracy, the better the model. For RankTopic, $\xi = 0.3$ is only shown. TR represents topical ranking model.

RankTopic are comparable with one of the state-of-the-art link combined topic models RTM in both document clustering and classification. Of course, to achieve the best performance, parameter ξ should be set properly, empirically ξ can be set to values close to 0.5.

D. Topic Interpretability

Lastly, we study the topic interpretability of RankTopic. Topics detected by topic models are represented as a distribution over words in the vocabulary. The detected topics can be interpreted as high level concepts from their top probability words. The more easier the topics can be interpreted as meaningful concepts, the better the detected topics. We define the degree of how easy a topic can be interpreted as a semantically meaningful concepts as *topic interpretability*.

However, the interpreting process of a topic can be rather complicated, which depends on the domain knowledge and comprehensive ability of an interpreter. Nevertheless, there exist some methods that try to evaluate the topic interpretability in a quantitative way. One such method is to use point-wise mutual information (PMI) [30] between pairs of words to evaluate the topic coherence. Higher PMI reflects better topic interpretability. In our experiments, we represent each topic by using their top 10 words and compute PMI between those words. The PMI of a topic is computed as the average PMI of all pairs of top probability words of that topic.

We first study the effect of parameter ξ on the topic interpretability of RankTopic. Taking ArnetMiner data as an example, Figure 11 illustrates the average and median

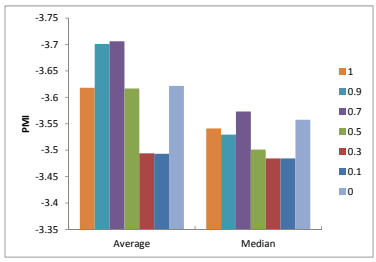


Figure 11. The average and median PMI values of topics detected by RankTopic with different ξ on the ArnetMiner data set. Notice that the y-axis is shown in a reverse order, so the lower the bar the better the topic interpretability.

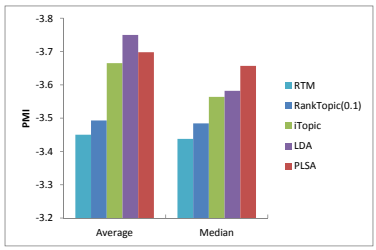


Figure 12. The average and median PMI values of topics detected by different topic models. The lower the bar, the better topic interpretability. For RankTopic, PMI for $\xi = 0.1$ is only shown.

PMI values of topics detected by RankTopic with some typical values of ξ. We see that when ξ is set to relatively low values, such as 0.1 and 0.3, the topic interpretability archives the best, while when ξ is set to 0, the topic interpretability becomes worse. The results are consistent with those of generalization performance, which suggests that there are correlation between the generalization performance and topic interpretability of topic models.

To compare the topic interpretability of different topic models, we also compute the average and median of PMI values of topics detected by the baseline models. Figure 12 presents the comparison results, from which we can see that RankTopic performs better than some baseline topic models and are slightly worse than RTM in terms of topic interpretability. Besides the quantitative evaluation of topic interpretability, we also compare the topics detected by RankTopic and one of the baseline models LDA in a manual way.

For example, Figure 13 shows one topic detected by LDA and two topics detected by RankTopic in ArnetMiner data. The titles for the topics are manually given out according to the semantic of the top 10 words. Topic 4 detected by LDA is interpreted as Language by us. However, this topic is actually a mixture of two concepts. One is programming language, which is indicated by red words. Another is natural language, which is indicated by blue words. The two concepts are well distinguished by RankTopic as two topics, Topic 3 (Programming) and Topic 10 (Semantic). From the experiments, we also find out that RankTopic clearly discriminates topic Architecture detected by LDA as

Topic 4
Language
language
knowledge
programming
domain
development
oriented
context
semantics
semantic
languages

(a) LDA

Topic 3	Topic 10
Programming	Semantic
program	knowledge
code	semantic
programs	document
java	text
programming	documents
execution	retrieval
language	ontology
source	mining
type	content
flow	task

(b) RankTopic

Figure 13. Example topics detected by LDA and RankTopic in ArnetMiner data set.

Computer Architecture and Service Oriented Architecture. Overall, all the 10 topics detected by RankTopic are easy to be interpreted to meaningfull research directions from the top probability words while some topics detected by LDA are difficult to be interpreted.

VI. Conclusion and Future Work

In this paper, we propose ranking based topic model called RankTopic for short, which incorporates link based ranking into topic modeling. To validate the effectiveness of RankTopic, we have studied the performance of RankTopic in various aspects, including generalization performance, document clustering and classification, and topic interpretability and have compared RankTopic with traditional topic models, PLSA and LDA, and link combined topic models, iTopic and RTM. Especially, we have investigated the model on a wide range of typical balancing parameter values and find out that RankTopic is sensitive to that parameter and it is indeed necessary to introduce such parameter to combat link noises. Extensive experiments show that when setting a proper balancing parameter ξ RankTopic performs consistently better than PLSA, LDA and iTopic, and is comparable with RTM in all the aspects on three public paper citation data sets and one twitter data set. As future works, we will study how RankTopic can benefit other applications, such as document retrieval and recommendation. Furthermore, we will explore how to use the similar idea of RankTopic to further improve some extended topic models, such as author-topic model and dynamic topic model.

Acknowledgements. This work is supported by National Natural Science Foundation of China under Grants 70771043 and 61173170, National High Technology Research and Development Program of China under Grant 2007AA01Z403, and Innovation Fund of Huazhong University of Science and Technology under Grants 2012TS052 and 2012TS053. We sincerely thank the anonymous reviewers for their very comprehensive and constructive comments.

References

[1] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: extraction and mining of academic social networks." in *KDD*, 2008, pp. 990–998.

[2] T. Hofmann, "Probabilistic latent semantic analysis," in *UAI*, 1999, pp. 289–296.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *JMLR*, vol. 3, no. Jan, pp. 993–1022, 2003.

[4] L. Nie, B. D. Davison, and X. Qi, "Topical link analysis for web search," in *SIGIR*, 2006, pp. 91–98.

[5] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer Networks*, vol. 30, no. 1-7, pp. 107–117, 1998.

[6] N. Ma, J. Guan, and Y. Zhao, "Bringing pagerank to the citation analysis," *Information Processing and Management*, vol. 44, no. 2, pp. 800–810, 2008.

[7] Y. Sun, J. Han, J. Gao, and Y. Yu, "iTopicmodel: Information network-integrated topic modeling," in *ICDM*, 2009, pp. 493–502.

[8] Q. Mei, D. Cai, D. Zhang, and C. Zhai, "Topic modeling with network regularization," in *WWW*, 2008, pp. 101–110.

[9] J. Chang and D. M. Blei, "Hierarchical relational models for document networks," *The Annals of Applied Statistics*, vol. 4, no. 1, pp. 124–150, 2010.

[10] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *ICML*, 2006, pp. 113–120.

[11] T. Iwata, T. Yamada, Y. Sakurai, and N. Ueda, "Online multiscale dynamic topic models," in *KDD*, 2010, pp. 663–672.

[12] X. Wang and A. McCallum, "Topics over time: a non-markov continuous-time model of topical trends," in *KDD*, 2006, pp. 424–433.

[13] Z. Yin, L. Cao, J. Han, C. Zhai, and T. S. Huang, "LPTA: A probabilistic model for latent periodic topic analysis," in *ICDM*, 2011, pp. 904–913.

[14] M. Rosen-Zvi, T. L. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *UAI*, 2004, pp. 487–494.

[15] R. Nallapati, A. Ahmed, E. P. Xing, and W. W. Cohen, "Joint latent topic models for text and citations," in *KDD*, 2008, pp. 542–550.

[16] C. X. Lin, Q. Mei, J. Han, Y. Jiang, and M. Danilevsky, "The joint inference of topic diffusion and evolution in social communities," in *ICDM*, 2011, pp. 378–387.

[17] C. Wang and D. M. Blei, "Collaborative topic modeling for recommending scientific articles," in *KDD*, 2011, pp. 448–456.

[18] R. Nallapati, D. A. McFarland, and C. D. Manning, "Topicflow model: Unsupervised learning of topic-specific influences of hyperlinked documents," *Journal of Machine Learning Research - Proceedings Track*, vol. 15, pp. 543–551, 2011.

[19] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM*, vol. 46, no. 5, pp. 604–632, 1999.

[20] J. Tang, J. Zhang, R. Jin, Z. Yang, K. Cai, L. Zhang, and Z. Su, "Topic level expertise search over heterogeneous networks," *Machine Learning*, vol. 82, no. 2, pp. 211–237, 2011.

[21] Y. Sun, Y. Yu, and J. Han, "Ranking-based clustering of heterogeneous information networks with star network schema," in *KDD*, 2009, pp. 797–806.

[22] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu, "Rankclus: integrating clustering with ranking for heterogeneous information network analysis," in *EDBT*, 2009, pp. 565–576.

[23] T. Yang, R. Jin, Y. Chi, and S. Zhu, "Combining link and content for community detection: A discriminative approach," in *KDD*, 2009, pp. 927–935.

[24] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang, "Mining topic-level influence in heterogeneous networks," in *CIKM*, 2010, pp. 199–208.

[25] G. Heinrich, "Parameter estimation for text analysis," University of Leipzig, Tech. Rep., 2008.

[26] J. Shi and M. J., "Normalized cuts and image segmentation," *PAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[27] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys Rev E Stat Nonlin Soft Matter Phys*, vol. 69, no. 2, 2004.

[28] R. Cilibrasi and P. M. B. Vitányi, "Clustering by compression," *IEEE Transactions on Information Theory*, vol. 51, no. 4, pp. 1523–1545, 2005.

[29] U. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, pp. 395–416, 2007.

[30] D. Andrzejewski and D. Buttler, "Latent topic feedback for information retrieval," in *KDD*, 2011, pp. 600–608.