Data Science and Engineering (2020) 5:27–47
https://doi.org/10.1007/s41019-019-00113-0

Bi-Labeled LDA: Inferring Interest Tags for Non-famous Users in Social Network

Jun He[1] · Hongyan Liu[2] · Yiqing Zheng[1] · Shu Tang[1] · Wei He[1] · Xiaoyong Du[1]

Received: 15 May 2019 / Revised: 10 September 2019 / Accepted: 11 November 2019 / Published online: 29 November 2019

Abstract
User tags in social network are valuable information for many applications such as Web search, recommender systems and online advertising. Thus, extracting high quality tags to capture user interest has attracted many researchers' study in recent years. Most previous studies inferred users' interest based on text posted in social network. In some cases, ordinary users usually only publish a small number of text posts and text information is not related to their interest very much. Compared with famous user, it is more challenging to find non-famous (ordinary) user's interest. In this paper, we propose a probabilistic topic model, *Bi-Labeled LDA,* to automatically find interest tags for non-famous users in social network such as Twitter. Instead of extracting tags from text posts, tags of non-famous users are inferred from interest topics of famous users. With the proposed model, the formulation of social relationship between non-famous users and famous user is simulated and interest tags of famous users are exploited to supervise the training of the model and to make use of latent relation among famous users. Furthermore, the influence of popularity of famous user and popular tags are considered, and tags of non-famous users are ranked based on random walk model. Experiments were conducted on Twitter real datasets. Comparison with state-of-the-art methods shows that our method is more superior in terms of both ranking and quality of the tagging results.

Keywords Topic model · LDA · Labeled LDA · Social network · Social tagging · Random walk

Abbreviations

LDA	Latent Dirichlet Allocation
Labeled LDA	Labeled Latent Dirichlet Allocation
Bi-LDA	Bidirectional LDA
Bi-Labeled LDA	Bidirectional Labeled LDA
DCG	Discounted Cumulative Gain

1 Introduction

Online social networking platforms like Twitter have become a mainstream medium, attracting millions of people spending their time there every day. Capturing interest and preference of users in these platforms is very important for many applications such as recommender system, personalized search and online advertising, besides social networking service itself. Tagging is one effective way to describe user's preference interest. Some social networking platforms such as Twitter don't provide chance for users to tag themselves. Others allow users to provide tags to describe themselves, but these tags are usually ambiguous, trivial, inadequate or even plain false [7]. Therefore, how to tag users automatically and accurately becomes a hot research topic in recent years. Non-famous[1] users in social network platforms are usually less active and provide less information than famous users, making it more challenging to find appropriate tags for them. Therefore, in this paper, we focus on study how to find interest tags for non-famous users in Social network platforms such as Twitter.

A key challenge of solving this problem is how to accurately infer the topics of interest for a user u. Most prior studies attempted to infer the topics of interest from the tweet content posted or retweeted by u in Twitter, mainly using topic models such as LDA and Labeled Latent Dirichlet Allocation (labeled LDA) [13]. Labeled LDA is one of

✉ Hongyan Liu
hyliu@tsinghua.edu.cn

[1] Key Laboratory of Data Engineering and Knowledge Engineering, MOE, Renmin University of China, Beijing 100872, China

[2] Department of Management Science and Engineering, Tsinghua University, Beijing 100084, China

[1] We define users who are followed by less than 2000 users as non-famous users, and users who are followed by more than 2000 users as famous users [6, 7].

the most competitive models for solving this problem using tweet content information. Some other approaches use both the tweet content and the social relationship information, mining users' topics of interest from tweets and re-ranking users' interests based on underlying Social network [12, 18]. However, people often post interest-unrelated tweets about their lives [4, 17, 19]. Therefore, tweets users published usually cannot reflect or cover all topics of their interests.

To address these problems, Bhattacharya et al. [1] proposed a method that first determines the topical expertise of popular Twitter users based on their Twitter Lists features[2] and then transitively infers the interests of the users who follow them. Through this approach, tags extracted for popular user are of high quality. But many popular users cannot get tags and non-popular users usually get popular tags. Our experiments show that it usually recommends popular tags such as "celeb," "news," "media," for non-famous users. Ding et al. [4] proposed a method to extract interest tags from Twitter user biographies, which heavily depends on the availability of users' biographies. Lappas et al. [7] proposed to use traditional LDA model to find the famous aspects of popular Twitter users based on their published tweets and to infer tags of non-famous users based on their *following* relationship with the popular users. In this work, words of tweets were extracted as tags, which often have low level of generalization. In addition, every popular user was regarded as a unique id in the LDA model, which ignored the relations between popular users and undermined the effectiveness of the proposed method.

In this paper, we study two mining problems: how to extract interest tags for non-famous users based on social relationship among users in social network, without using tweet information, and how to rank tags of each non-famous user, capturing the importance of different tags. In particular, we extend traditional topic model LDA to model non-famous user's following behavior, making use of famous user's tag information simultaneously. People usually follow a famous user for personal interest reason. Therefore, famous users share quite the same interests with the non-famous users who follow them, which is called *homophily* in [22]. Based on this phenomenon, famous user's interest information is incorporated into traditional topic model LDA to serve as labels of documents and labels of words, and a probabilistic topic model called Bi-Labeled LDA is developed based on two basic intuitions. To further enhance the performance of this model, we improve it by relaxing the assumption that a famous user is followed due to one topic of interest and taking high popularity issues into account. Based on the result of this topic model, a random walk model is proposed to further rank tags for each non-famous user, utilizing social relationship among non-famous users. Ultimately based on these model results, we can output a ranked list of tags to describe each user's interests.

The major contributions we make in this paper are as follows:

- We propose a new topic model, *Bi-Labeled LDA*, to model the process in which non-famous user follows famous users and infer tags for non-famous users effectively. Comparing to existing model, it takes the relation between famous users into consideration, incorporating more supervision information into traditional LDA. *Bi-Labeled LDA* is further improved to address two issues: strong assumption behind LDA and high popularity of topic and famous user.
- A Random Walk model is proposed to rank the tags inferred through *Bi-Labeled LDA*, adjusting the importance of unpopular tags among famous users.
- We conducted comprehensive experiments on real dataset and compared the interest tags found based on the proposed models and state-of-the-art approaches. We find that interest tags extracted by our methodology are far superior to others either in accuracy and have better generalization.

The rest of the paper is organized as follows: In Sect. 2, related work is discussed. We describe all problems and clear the definitions in Sect. 3. Then, the proposed *Bi-Labeled LDA* model and its extensions to infer interest tags for non-famous users are presented in Sect. 4. And, the method to find the interest tags of famous users is introduced in Sect. 5. In Sect. 6, experimental setup and results are described. Finally, conclusions are drawn in Sect. 7.

2 Related Work

Closely related existing work can be categorized into two groups: One group of work mainly utilizes users' tweet information to extract their interest topics, and the other group utilizes other kinds of information such as biography and social information to infer their interests.

Most prior studies attempted to mine user interests from the tweets posted or retweeted, mainly using topic models such as LDA. Xu et al. [20] proposed a modified author-topic model named twitter-user model to discover users' topics of interest by filtering out interest-unrelated tweets from the aggregated user tweets. For each tweet, they introduced a latent variable to indicate whether it is related to its author's interests. Zhao et al. [23] developed a new topic

[2] Twitter introduced Lists in late 2009, to help users organize their followings (i.e., the people whom a user follows). When creating a List, a user typically provides a List name (free text, limited to 25 characters) and optionally adds a List description.

model named Twitter-LDA to improve the quality of topics by restricting each tweet to one topic and a common background topic. Quercia et al. [11] inferred users' topics of interest with a supervised topic model, Labeled Latent Dirichlet Allocation (*Labeled LDA*), and showed it to be more effective than LDA. Specifically, *Labeled LDA* uses the same underlying mechanisms as traditional LDA, but each topic is seeded with a label, to help anchor the topic extraction process. Quercia et al. labeled each Twitter user using some text classification APIs, while Ottoni et al. [10] selected the 300 most common hashtags from all the tweets as topic labels. Michelson and Macskassy [9] proposed to find user's interest with entity categories through extracting entities from tweets and categorizing entities based on Wikipedia. Some approaches used both tweets and network information, mining users' topics of interest from tweets and then re-ranking users' interests based on underlying social network using technique such as Random Walk [12, 18].

After all, all the methods mentioned above rely on the tweet content, but Twitter users often post tweets about their daily lives or have conversation with their friends, which are usually not related to their interests [4, 17, 19], and 82.2% Twitter users post less than 100 tweets per year [7], which both make it difficult to infer meaningful topics from tweets. To address this problem, most studies focused on users' other features, such as biographies and network information, and incorporated extra information such as Wikipedia and human effort [1, 4, 7, 8]. Bhattacharya et al. [1] first deduced the topical expertise of famous Twitter users based on their Twitter Lists features and then transitively inferred the interests of the users who follow them. Although their approach is very effective for deducing the topical expertise of famous users, it doesn't perform well for non-popular users. Our experiments show that it always recommends famous tags such as "celeb," "news," and "media," to non-famous users. Ding et al. [4] extracted interest tags from Twitter user biographies, with a sequential labeling model based on automatically constructed labeled data. However, their approach heavily depends on the availability of users' biographies, and as a matter of fact only 22% of Twitter users have a biography on their profile [13] and Ding et al. revealed that only 28.8% of biographies contain meaningful interest tags. Then, even in the most ideal case, they are only able to recommend interest tags for 6.336% users in Twitter. Lim and Datta [8] introduced a method to find a user's interest through classifying their celebrity followings into categories. Celebrity is categorized through extracting keywords from occupation or the first paragraph of description text presented on Wikipedia and mapping from the extracted keywords to category. This method depends on information presented on Wikipedia for only real-life celebrities, and how to build the mapping from keywords to categories is not solved. Lappas et al. [7] inferred the famous aspects of popular Twitter users using two standard LDA models: one for the generative process of non-famous users' followings behavior, where every popular Twitter user is regarded as a word token, and the other for the generative process of tweets. Finally, words of tweets were extracted as tags, which often have low level of generalization. In addition, every popular user was regarded as a unique id in the LDA model, which ignored the co-occurrence information among popular users.

In this paper, we propose a model to find interest tags for non-famous users based on the underlying social network in Twitter, without using tweet text information. In addition, by mapping each famous user with a tag set and modeling the user's following behavior, it takes relations between famous users into consideration. Besides social relationship between non-famous user and famous user, relationship between non-famous users is also used to rank tags inferred through the topic model, improving performance further.

3 Problem Definition

Given a set of users on social network platforms such as Twitter, for each user u, we have all of its followings, i.e., the users u follows, and the *List* information of the followings, including name and description of each list. A user may be a person, a company, an organization, etc. Among these users, we define those who are followed by less than 2000 users as non-famous users, and users who are followed by more than 2000 users as famous users [6, 7]. Then, we split these users into two sets: a set U of non-famous users and a set V of famous users.

Given the above information, in this paper we have the following three mining tasks:

Mining task 1 Given a set V of famous users and their followers, we want to extract interest tags for each famous user based on List information.

As a result of task 1, we obtain a set K of tags, each of which represents an interest topic of famous users. Let the set $\mathbf{K} = \{t_1, t_2, \ldots, t_{|K|}\}$. For each famous user v, we describe its interests by a set of tags, denoted by a binary vector, $\mathbf{T}^{(v)} = \left(\mathbf{T}_1^{(v)}, \ldots, \mathbf{T}_{|K|}^{(v)}\right)$, where each $\mathbf{T}_k^{(v)} \in \{0, 1\}$, $\mathbf{T}_k^{(v)} = 1$ if user v has tag t_k; otherwise, $\mathbf{T}_k^{(v)} = 0$.

Mining task 2 Give a set U of non-famous users, a set V of famous users with tags, and following relationship between these two sets of users, we want to infer a set of tags for each non-famous user to represent their interests.

Mining task 3 Give a set U of non-famous users, each associated with a set of tags, we want to rank the tags so that the higher the rank, the more possible the user is interested in the topic the tag represents.

We describe how to fulfill task 1 in Sect. 4 and the other two tasks in Sect. 5. Our major contribution focuses on task 2 and task 3. To make the procedure more understandable, we describe task 1 first. We will use the expression of topic and tag interchangeably hereafter according to context. A tag represents an interest topic of both famous users and non-famous users.

4 Extracting Interest Tags of Famous Users

As famous users are usually active users, with more activities and more information posted every day, there are many ways to extract interest tags for them. In this paper, we take advantage of *List* in Twitter to do that. In Twitter, *List* is introduced to help users organize their followings. A user can create a List, specify a List name and an optional description, and then add some of his followings to this List. Usually, a famous Twitter user is a member of many Lists. For instance, Barack Obama is a member of Lists such as "politics", "government", "celeb", "leader", etc.

Ghosh et al. [5] proposed a method to discover the topical expertise of famous users utilizing Twitter List names and descriptions. We adopt this method and improved it to tag famous users. We adopt TweetNLP[3] to perform POS tagging for text information. Before extracting tags, we filter each list by using the GNU Aspell dictionary. GNU Aspell is a free and open-source spell checker, which can determine Out of vocabulary (OOV) tokens. It includes support for using multiple dictionaries and can remove noisy words efficiently. After that, we normalize the lists by using normalization system.[4] This normalization system detects and expands word tokens not in standard type including abbreviations and acronyms. Then, we merge synonyms based on WordNet. Finally, we remove stop words and perform stemming by using the Porter stemming algorithm. After these preprocessing steps, we extract tags of famous users based on the method used by Bhattacharya et al. [1, 5]. According to this method, to find interest tags of a famous Twitter user v, we first collect the Lists which have v as a member, and then extract frequently occurring terms (unigrams and bigrams which are identified as *nouns* or *adjectives*) from the List names and descriptions. For each term, we count its frequency, the number of times it occurs in the list names or descriptions. In particular, if term t has frequency no less than 10, we identify v as an expert on a topic t, and we regard t as a *tag* of user v. As a result, each famous user who is member of at least one List is temporarily tagged by the set of terms extracted. Then, we retain those tags which occur in more than 1% users' tags. Applying this method on the experimental dataset, after this step, we get a set of meaningful and qualified tags, and we keep only these tags to infer non-famous user's tags.

Through this method, in our experimental dataset, 61.1% of famous users have at least one tag. To improve this method and to infer tags for more famous user, first, for each non-famous user u, we form a temporary tag set by obtaining the union of tag sets of famous users u follows. For those famous users for whom we cannot infer tags, we then tag them based on non-famous user's temporary tag set. For a famous user v, let $f(v)$ be the set of non-famous users who follow v, and $CT(u)$ be the temporary tag set of non-famous user u, then user v's tag set, denoted by $tag(v)$, is inferred according to Eq. (1):

$$\text{tag}(v) = \bigcap\nolimits_{u \in f(v)} CT(u) \tag{1}$$

That is to say, the intersection of non-famous followers' temporary tag set is regarded as the famous user's tag set. In this way, 93.4% of famous users finally have tags.

5 Inferring Tags for Non-famous Users

Based on the tags famous users have, in this section we introduce our methods to infer tags of non-famous users based on social relationship information.

Suppose each famous user has a set of tags, each of which represents an interest topic which they are an expert at or famous in. Meanwhile, they are usually more famous in some topics than in others. For example, Lance Armstrong[5] has two tags: cyclist and cancer survivor. And he is more famous as a world-class cyclist than as a cancer survivor. A non-famous user follows a famous user due to some of the topical expertise of the famous user. For example, a user follows Lance Armstrong because he is an expert or famous in cycling. Therefore, when a non-famous user u follows a famous user v, we assume that user u has different levels of interest in each topic represented by a tag of user v. However, we only observe that user u follows user v; it is not easy to find the different influence of user v's different interest topics on user u.

In order to figure out the reason why a non-famous user u follows a famous user v, we have the following intuitions:

[3] http://www.ark.cs.cmu.edu/TweetNLP.

[4] https://github.com/EFord36/normalise.

[5] https://twitter.com/lancearmstrong.

- *Intuition 1* If a non-famous user u follows more users who are famous in topic a than the ones who are famous in topic b, u follows a famous user v who is famous both in topics a and b more because of interest in topic a than in topic b.

  For example, suppose user u follows ten famous users. We count tags of these famous users and get three tags with counts: *entertainment* (6), *business* (1), and *food* (5). For a particular famous user v with tags *entertainment* and *business*, we think user u follows v more because of interest in topic *entertainment* than in topic *business*.
- *Intuition 2* If a famous user v is followed by more non-famous users with interest in topic a than in topic b, then v is followed by a non-famous user u more due to u's interest in topic a than in topic b.

  For example, suppose a famous user v with tags *entertainment* and *business* is followed by ten non-famous users. Among these non-famous users, six have interest in topic *entertainment*, one has interest in *business*, and five have interest in *food*. Then, non-famous user u follows v more because of his/her interest in topic *entertainment* than topic *business*.

Based on the observations and intuition discussed above, we propose a modified topic model, *Bi-Labeled LDA*, to model the generative process of non-famous users' followings behavior. In this model, we assume a non-famous user u follows a famous user v because of one topic of interest. Different from the model proposed by Lappas et al. [7], tags of famous users are exploited to supervise the learning of the model, linking different famous users through tags. In this model, we first only make use of the following relationship between non-famous users and famous users. We exclude social relationship between non-famous users in this step because we want to eliminate noise as far as possible. Existing study [22] has shown that sometimes a non-famous user follows another non-famous user may be due to the fact that they are offline friends, families, or just following each other back, not for real interest.

This model is further improved by relaxing the assumption that a famous user is followed because of one topic of interest. The following behavior may be owing to some topics or popularity of the famous user.

Based on *Bi-Labeled LDA*, we find a set of tags for each non-famous user with each tag representing an interest topic. But a user may be more interested in some topics than others. Therefore, we take advantage of following relationship between non-famous users to rank each user's tags in the end.

In the following, we first introduce the basic model of *Bi-Labeled LDA* and its extension and then describe the ranking model.

5.1 Bi-Labeled LDA

To perform mining task 2, we propose a probabilistic topic model, *Bi-Labeled LDA*, to model the process in which non-famous users follow famous users in social networking platforms such as Twitter. This model is an improvement in traditional topic model LDA [2], which is originally proposed to model the generative process of a document. According to LDA, each word of a document is generated through two steps: first pick a topic based on document-specific topic distribution, and then, pick a word based on word distribution of the picked topic, under the assumption that a document is a mixture of latent topics. To model the user's following behavior in social network platforms, we have similar assumption that each user has a mixture of latent interest topics. Each tag of a famous user represents one interest topic. To get information about one topic, a user chooses famous users who has the same topic interests to follow. Therefore, the set of famous users a user u follows reflects u's latent interests. In analogy to document generation, each non-famous user u is regarded as a document, consisting of a set of famous users who are followed by user u. Hence, each followed famous user corresponds to a word of the document. For simplicity, famous users followed by a user u are called u's followings. Informally, to generate a document, a non-famous user u first picks a topic from his personal distribution of interest topics and then picks a famous user in that topic based on the topic's distribution over all the famous users. We call this process ***following behavior generative process***.

Formally, given the set $\boldsymbol{U}$ of all non-famous users who follow a set $\boldsymbol{V}$ of famous users, we can represent each non-famous user u by a bag of famous users u follows, denoted by $\boldsymbol{V}^{(u)} = \left(V_1^{(u)}, \ldots, V_{N_u}^{(u)}\right)$, where each $V_i^{(u)} \in \boldsymbol{V}$. Here N_u is the number of famous users followed by users u. Through the method introduced in Sect. 4, we extract a set of tags, $\mathbf{K} = \left\{t_1, t_2, \ldots, t_{|K|}\right\}$, and each famous user v's tag set is denoted by a vector, $\mathbf{T}^{(v)} = \left(\mathbf{T}_1^{(v)}, \ldots, \mathbf{T}_{|K|}^{(v)}\right)$, where each $\mathbf{T}_k^{(v)} \in \{0, 1\}$, $\mathbf{T}_k^{(v)} = 1$ if user v has tag t_k; otherwise, $\mathbf{T}_k^{(v)} = 0$.

To find tags of a non-famous user u, we first build a ***candidate tag set for u***, which is the union of tag sets of famous users u follows and denoted by a vector, $\Lambda^{(u)} = \left(\Lambda_1^{(u)}, \ldots, \Lambda_{|K|}^{(u)}\right)$, where each $\Lambda_k^{(u)} \in \{0, 1\}$; $\Lambda_k^{(u)} = 1$ if tag t_k is a tag of user $v \in V^{(u)}$; otherwise, $\Lambda_k^{(u)} = 0$.

As famous users have tags, that is to say, each famous user is labeled with a set of tags, we want to use this label information to supervise the generative process of non-famous user's *following* behavior and use tags in K to express non-famous user's interest. Based on this idea, we propose the

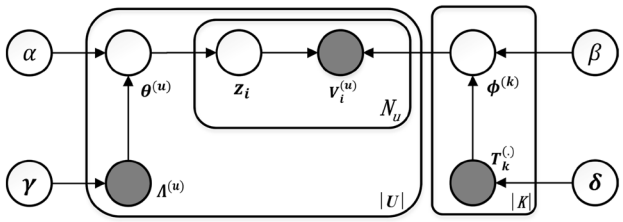

Fig. 1 Graphical model of Bi-Labeled LDA

Bi-Labeled LDA model. Its graphical representation is illustrated in Fig. 1.

The symbols used in the model are summarized in Table 1.

The top half part of the graphical model is the same as the standard LDA, which says that for user u, each famous user followed by user u is generated by first picking a topic Z_i based on $\boldsymbol{\theta}^{(\boldsymbol{u})}$ and then picking a famous user $V_i^{(u)}$ based on $\boldsymbol{\phi}^{(\boldsymbol{k})}$.

What is different is the lower half part. We think a user u's topics come from the tags of famous users followed by user u. Therefore, a user's topic distribution is restricted by a label prior γ. Similarly, for each topic t_k, its famous user distribution is restricted by a label prior δ. Specifically, user u's topic distribution is restricted to be only over user u's candidate tag set (while each tag is regarded as a possible latent topic), and the famous user distribution of a topic t_k is restricted to be only over those famous users who have this topic (i.e., the famous user who has tag t_k).

In other words, unlike traditional LDA, *Bi-Labeled LDA* defines a one-to-one correspondence between latent topics and tags. Every document is restricted to those topics that correspond to its candidate tag set. Meanwhile, every famous user is restricted to be generated (followed) from these topics, i.e., every topic can only have famous users associated with the same topic (tag). In this way, we incorporate supervision into traditional LDA and, meanwhile, take advantage of the relation among famous users who have same tags.

Let $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_{|K|}\}^T$ and $\boldsymbol{\beta} = \{\beta_1, \ldots, \beta_{|V|}\}^T$ be the Dirichlet smoothing parameters for topics and words, respectively, $\boldsymbol{\delta} = \{\delta_1, \ldots, \delta_{|K|}\}^T$ and $\boldsymbol{\gamma} = \{\gamma_1, \ldots, \gamma_{|K|}\}^T$ be the label priors for topic and non-famous users, respectively, $\boldsymbol{\theta}^{(\boldsymbol{u})} : \left\{\theta_k^{(u)} = p(t_k|u), \forall t_k \in \boldsymbol{K}\right\}$ be non-famous user u's topic vector, $\boldsymbol{\phi}^{(\boldsymbol{k})} : \left\{\phi_v^{(k)} = p(v|t_k), \forall v \in \boldsymbol{V}\right\}$ be the topic t_k distribution over famous users. Let $\boldsymbol{L}^{(\boldsymbol{u})}$ and $\mathbf{M}^{(\boldsymbol{k})}$ be two matrices used to constrain the topics user u could have and the topics v could belong to, respectively.

In order to restrict $\boldsymbol{\theta}^{(\boldsymbol{u})}$ to be defined only over the topics that correspond to u's candidate tag set represented by $\boldsymbol{\Lambda}^{(u)} = \left(\Lambda_1^{(u)}, \ldots, \Lambda_{|K|}^{(u)}\right)$, we define a tag projection matrix $\boldsymbol{L}^{(\boldsymbol{u})}$ of size $|K| \times |K|$ for each non-famous user u. For each row $i \in \{1, \ldots, |K|\}$ and column $j \in \{1, \ldots, |K|\}$:

$$L_{ij}^{(u)} = \begin{cases} \Lambda_i^{(u)} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

Then, $\boldsymbol{\alpha}^{(\boldsymbol{u})}$ is computed according to Eq. (2):

$$\boldsymbol{\alpha}^{(\boldsymbol{u})} = \boldsymbol{L}^{(\boldsymbol{u})} \times \boldsymbol{\alpha} = (\alpha_1^{(u)}, \ldots, \alpha_{|K|}^{(u)})^T \tag{2}$$

In other words, $\alpha_k^{(u)}$ is equal to α_k if and only if $\Lambda_k^{(u)}$ is 1, and 0 otherwise. Clearly, the topics of user u are constrained to its candidate tag set.

For example, suppose $|\boldsymbol{K}| = 4$, a non-famous user u's candidate tag set is denoted by vector $\boldsymbol{\Lambda}^{(u)} = (1, 0, 0, 1)$, then $\boldsymbol{L}^{(\boldsymbol{u})}$ **and** $\boldsymbol{\alpha}^{(\boldsymbol{u})}$ are shown as below:

Table 1 Symbols used in Bi-Labeled LDA

Symbol	Description
U	The set of non-famous users
$V, V_i^{(u)}$	The set of famous users and the ith famous user followed by user u
K	The set of topics (tags) famous users have
Z_i	The ith topic
N_u	The number of famous users followed by user u
$\mathrm{T}^{(v)}$	The famous user v's tag set
$\Lambda^{(u)}$	A binary vector to represent u's candidate tags
α, β	Dirichlet smoothing parameters of topics and words, respectively
δ, γ	Label priors for topics and non-famous users, respectively
$\theta^{(u)}$	The non-famous user u's topic distribution
$\phi^{(k)}$	The topic t_k's distribution over famous users
$\alpha^{(u)}$	The Dirichlet smoothing parameters for non-famous user u
$\beta^{(k)}$	The Dirichlet smoothing parameters for topic k

$$\boldsymbol{L}^{(\boldsymbol{u})} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\boldsymbol{\alpha}^{(\boldsymbol{u})} = \boldsymbol{L}^{(\boldsymbol{u})} \times \boldsymbol{\alpha} = \{\alpha_1, 0, 0, \alpha_4\}^T$$

Similarly, we define a matrix $\mathbf{M}^{(\boldsymbol{k})}$ of size $|V| \times |V|$ for each topic t_k. For each row $i \in \{1, \ldots, |V|\}$ and column $j \in \{1, \ldots, |V|\}$:

$$M_{ij}^{(k)} = \begin{cases} \mathrm{T}_k^{(i)} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Similarly, $\boldsymbol{\beta}_{\mathrm{i}}^{(k)}$ is computed as:

$$\boldsymbol{\beta}^{(\boldsymbol{k})} = \boldsymbol{M}^{(\boldsymbol{k})} \times \boldsymbol{\beta} = \left(\beta_1^{(k)}, \ldots, \beta_{|V|}^{(k)}\right)^T \tag{4}$$

In other words, $\boldsymbol{\beta}_{\mathrm{i}}^{(k)}$ is equal to β_i if $\mathrm{T}_k^{(i)}$ is 1 (i.e., famous user i can belong to topic t_k), and 0 otherwise. Clearly, the topics a famous user can belong to are constrained to its associated tag set.

The generative process behind model *Bi-Labeled LDA* is shown below.

Bi-Labeled LDA: Generative process

For each famous user v:
  For each topic $t_k \in K$:
    Generate $T_k^{(v)} \epsilon \{0,1\} \sim Bernoulli(\boldsymbol{\delta_k})$
For each topic $t_k \in K$
  Compute $\boldsymbol{\beta}^{(\boldsymbol{k})} = \boldsymbol{M}^{(\boldsymbol{k})} \times \boldsymbol{\beta}$
  Generate $\boldsymbol{\phi}^{(\boldsymbol{k})} \sim Dir(\boldsymbol{\beta}^{(\boldsymbol{k})})$
For each user $u \in U$:
  For each topic $t_k \in K$
    Generate $\Lambda_k^{(u)} \epsilon \{0,1\} \sim Bernoulli(\boldsymbol{\gamma_k})$
  Compute $\boldsymbol{\alpha}^{(\boldsymbol{u})} = \boldsymbol{L}^{(\boldsymbol{u})} \times \boldsymbol{\alpha}$
  Generate $\boldsymbol{\theta}^{(\boldsymbol{u})} \sim Dir(\boldsymbol{\alpha}^{(\boldsymbol{u})})$
  For each famous user i whom user u follows
    Generate $z_i \sim Mult(\boldsymbol{\theta}^{(\boldsymbol{u})})$
    Generate $V_i^{(u)} \sim Mult(\boldsymbol{\phi}^{(\boldsymbol{z_i})})$

5.2 Learning and Inference

Similar to standard LDA, we learn $\boldsymbol{\theta}^{(\boldsymbol{u})}$ and $\boldsymbol{\phi}^{(\boldsymbol{k})}$ using collapsed Gibbs sampling [14]. The final sampling update equation for picking a topic to explain why user u follows user v is given in Eq. (5), assuming that v is the mth famous user in u's *following* list. Equations (6) and (7) are used to estimate $\boldsymbol{\theta}^{(\boldsymbol{u})}$ and $\boldsymbol{\phi}^{(\boldsymbol{k})}$.

$$\begin{aligned} \mathrm{p}(z_{u,m}|.) \propto & \mathrm{p}\left(z_{u,m} = t_k, w_{u,m} = v | \boldsymbol{Z}_{\neg(\boldsymbol{u},\boldsymbol{m})}, \boldsymbol{V}_{\neg(\boldsymbol{u},\boldsymbol{m})}, \boldsymbol{\alpha}^{(\boldsymbol{u})}, \boldsymbol{\beta}^{(\boldsymbol{k})}\right) \\ = & \frac{c_{k,u,*}^{-(u,m)} + \alpha_k^{(u)}}{c_{*,u,*}^{-(u,m)} + \alpha_*^{(u)}} \cdot \frac{c_{k,*,v}^{-(u,m)} + \beta_v^{(k)}}{c_{k,*,*}^{-(u,m)} + \beta_*^{(k)}} = \widehat{\theta_{k,\neg(\boldsymbol{u},\boldsymbol{m})}^{(u)}} \cdot \widehat{\phi_{v,\neg(\boldsymbol{u},\boldsymbol{m})}^{(k)}} \end{aligned} \tag{5}$$

$$\widehat{\theta_k^{(u)}} = \frac{c_{k,u,*} + \alpha_k^{(u)}}{c_{*,u,*} + \alpha_*^{(u)}} \tag{6}$$

$$\widehat{\phi_v^{(k)}} = \frac{c_{k,*,v} + \beta_v^{(k)}}{c_{k,*,*} + \beta_*^{(k)}} \tag{7}$$

where $c_{k,u,*}$ denotes the number of associations between a topic t_k and a non-famous user u, $c_{k,u,*}^{-(u,m)}$ denotes the count when we exclude the follow relation between a non-famous user u and a famous user v, and the symbol * denotes a summation over all possible subscript variables. Symbols are summarized in Table 2.

Note that in the above equations the topic prior $\boldsymbol{\alpha}^{(\boldsymbol{u})}$ is document specific, and the word prior $\boldsymbol{\beta}^{(\boldsymbol{k})}$ is topic specific. *Bi-Labeled LDA* captures the two intuitions discussed in the beginning of Sect. 4 well as explained below:

- If a non-famous user u follows more users who are famous in aspect x than the ones who are famous in aspect y, then $c_{x,u,*} > c_{y,u,*} \rightarrow \widehat{\theta_x^{(u)}} > \widehat{\theta_y^{(u)}}$, i.e., u follows a famous user v who is famous both in aspects x and y more because of interest in aspect x than in y.
- If a famous user v is followed by more non-famous users with interest in aspect x than that in aspect y, then $c_{x,*,v} > c_{y,*,v} \rightarrow \widehat{\phi_v^{(x)}} > \widehat{\phi_v^{(y)}}$, i.e., v is followed by a non-famous user u more due to u's interest in aspect x than in aspect y.

After the learning and inferring step, we obtained estimation of $\boldsymbol{\theta}^{(\boldsymbol{u})}$ and $\boldsymbol{\phi}^{(\boldsymbol{k})}$, which indicate a non-famous users u's topic distribution and the topic t_k's distribution over famous users, respectively. Since we map every topic to a tag, we then recommend the top ranked tags to a non-famous user according to their probability values in $\boldsymbol{\theta}^{(\boldsymbol{u})}$. As a result, each non-famous user is recommended a tag set, and we record each tag by a pair (tag, probability score), i.e., $\left(t_k, \theta_k^{(u)}\right)$.

5.3 Extension of *Bi-Labeled LDA*

In *Bi-Labeled LDA*, we assume a non-famous user follows a famous user only because of one topic. But it is possible that

Table 2 Symbols used for inferring model *Bi-Labeled LDA*

Symbol	Meaning
$\boldsymbol{Z}$	Denoting $\boldsymbol{Z} = (z_1, z_2, \ldots, z_{\|U\|})$, in which each $\boldsymbol{z_u} = (z_{u,1}, z_{u,2}, \ldots, z_{u,n_u})$ represents the topic assignment of user u's *followings*.
$\boldsymbol{W}$	Denoting $W = (w_1, w_2, \ldots, w_{\|U\|})$, in which each $\boldsymbol{w_u} = (w_{u,1}, w_{u,2}, \ldots, w_{u,n_u})$, representing n_u famous users whom user u follows.
$\boldsymbol{z_{u,m}}$	The topic assignment of the mth famous user followed by user u
$\boldsymbol{w_{u,m}}$	The *mth* famous user followed by user u
$\boldsymbol{c_{k,u,v}}$	The number of associations between a topic t_k and a famous user v followed by user u
$\boldsymbol{c_{k,u,*}}$	The number of associations in which u follows a famous user due to topic k. Symbol * denotes a summation over all possible subscript variables and here means all possible famous users followed by u
$\boldsymbol{c_{k,u,*}^{-(u,m)}}$	The number of times user u follows a famous user due to topic k excluding the current following behavior that non-famous user u follows the mth famous user

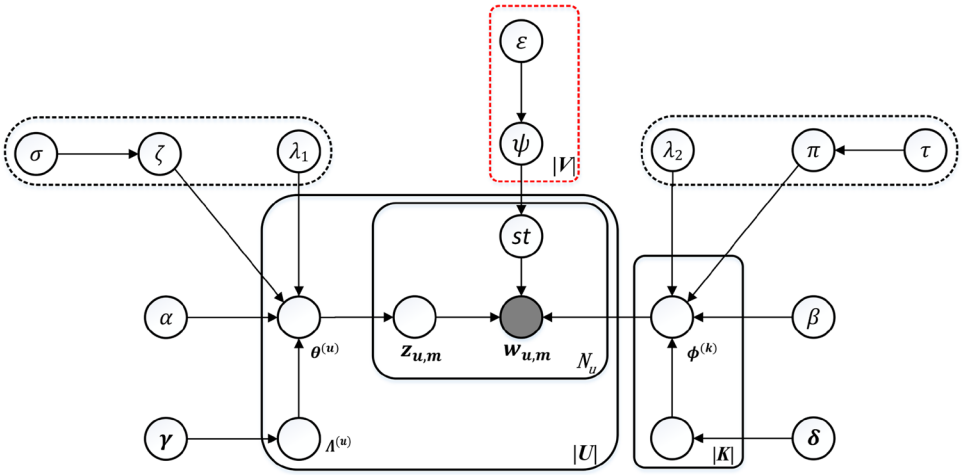

Fig. 2 Graphical model of *Bi-Labeled LDA2*

famous users are followed because of their multiple topics of interests. Meanwhile, the more topics a famous user is interested in, the more likely its popularity in some topics is overestimated.

Besides, high popularity issue [16] is ignored in *Bi-Labeled LDA*. In our problem setting, high popularity issue manifests in the following two aspects:

First, some famous users are more popular in some topics than others. The more popular a user v is, the more likely a user follows v not because of interest but because of v's popularity [18]. For example, you follow Barack Obama just because he is popular, not because you are interested in politics. Second, some topics are more popular than others. As shown in Eq. 6, more popular topics usually have bigger $c_{k,u,*}$, which may dominate u's other less popular interests.

Taking these points into consideration and inspired by models proposed in [3], we further improve *Bi-Labeled LDA* to solve these problems. To address the first issue, we assume some famous users are followed because of one topic and others may be because of more than one topic. We call the corresponding *following* relationship *one-topic following relation* and *multi-topics following relation*. Then, we can deal with these situations separately. Accordingly, in the *following behavior generative process*, there are two separate paths from which famous users are followed. This separation is expected to eliminate the cases in which a famous user is followed by a non-famous user owing to more than one topics, and ultimately help to generate topics with less bias. To do that, a new binary latent variable *st* (single topic) is introduced to indicate the path the famous user v comes from, where $st = 0$ means v comes from a "multi-topics" path and $st = 1$ means that v comes from a "one-topic" path. And in the sampling process, we do a "path labeling" as well as a "topic labeling" for a following behavior. The upper middle red dashed box component in Fig. 2 depicts how this idea is incorporated into the *Bi-Labeled LDA* model.

Taking high popularity issue into consideration, the reason why a famous user v is followed by a non-famous user u can be extended to include the following three situations:

- User u is interested in some topics in which user v is famous for.
- User v is very popular.
- One topic in which user v is famous is very popular.

Table 3 More symbols used in *Bi-Labeled LDA2*

Symbol	Meaning
$\boldsymbol{B}$	Denoting $\boldsymbol{B} = (b_1, b_2, \ldots, b_{\lvert U\rvert})$, in which each $\boldsymbol{b_u} = (b_{u,1}, b_{u,2}, \ldots, b_{u,n_u})$, representing the single topic label of each famous user followed by user u
λ_1	A concentration scalar constructing the global distribution of topics (tags)
λ_2	A concentration scalar constructing the global distribution of famous users
$\boldsymbol{\sigma_k}$	The smoothing parameter of topic k in the corpus
$\boldsymbol{\tau_v}$	The smoothing parameter of famous user v in the corpus
$\boldsymbol{\zeta}$	The prior observation of the topics in the corpus
$\boldsymbol{\pi}$	The prior observation of the famous users in the corpus
$\boldsymbol{\varepsilon}$	The Dirichlet smoothing parameter for the single topic labels of famous users
$\boldsymbol{\psi}$	The probability the famous users are picked to follow through single topic
$\boldsymbol{st}$	The indicator of whether a famous user is picked to follow through single topic
$\boldsymbol{c_{k,u,m,st}}$	The number of associations between a topic t_k and the mth famous user v followed by a non-famous user u when the single topic label is st

Based on these points, we update user's topic distribution by taking topic's global popularity into account. Meanwhile, we update topic's famous user distribution by taking famous user's popularity into consideration. This interpretation leads us to a polya-urn model [3] with two new components added to the model, the upper left and upper right parts in dashed boxes in Fig. 2, with meaning of symbols shown in Table 3.

For convenience, we call the model in Fig. 1 as ***Bi-Labeled LDA1*** and call this new model as ***Bi-Labeled LDA2***. The upper right dotted box shows the global popularity distribution of famous users, which consists of a multinomial distribution π, a Dirichlet prior τ, and a concentration scalar λ_2. Note that τ is a vector of length $|V|$, the number of unique famous users, and each element has a value of $\tau_v = \frac{f_v}{f_*}$, where f_v denotes the frequency of a famous user v in our dataset and f_* denotes a total frequency $\left(\sum_v f_v\right)$. As λ_2 works as a weight to the prior observation π, ϕ becomes similar to π when λ_2 has a high value. On the other hand, ϕ deviates from π when λ_2 has a low value. Things are the same for the left black dotted box component, which shows the global distribution of topics.

The detail of the red dotted box in Fig. 2 is as follows: The variable st follows a Bernoulli distribution ψ constrained by a Beta prior ε. When a famous user has many tags, the probability to be followed due to more than one topic becomes higher. Therefore, we pose an asymmetric prior according to the tag sets of famous users, which is mapped by a sigmoid function shown as follows:

$$\varepsilon_{w_{u,m},\mathrm{st}} = \begin{cases} 1 - \frac{1}{1+e^{-\frac{\left|\mathrm{Tag}(w_{u,m})\right| - \mathrm{median}_{v\in V}(|\mathrm{Tag}(v)|)}{C}}} & \text{if } st = 1 \\ 1 - \varepsilon_{w_{u,m},1} & \text{if } st = 0 \end{cases} \tag{8}$$

where $\left|\mathrm{Tag}\left(w_{u,m}\right)\right|$ denotes the number of tags in $w_{u,m}$'s tag set, $|\mathrm{Tag}(v)|$ denotes the number of tags famous user v has, median(x) returns the median of the set of x's values, and C is a scaling constant.

Finally, the topic assignment probability of this *Bi-Labeled-LDA2* model is updated as shown in Eqs. (9) and (10):

$$\begin{aligned} \mathrm{P}\left(\mathrm{st}_{u,m}|.\right) &\propto \mathrm{p}\left(\mathrm{st}_{u,m} = b, w_{u,m} = v|\boldsymbol{B}_{\neg(\boldsymbol{u},\boldsymbol{m})}, \boldsymbol{W}_{\neg(\boldsymbol{u},\boldsymbol{m})}, \boldsymbol{\varepsilon}\right) \propto \frac{c^{-(u,m)}_{*,*,v,\mathrm{b}} + \varepsilon_{v,\mathrm{b}}}{c^{-(u,m)}_{*,*v,*} + \varepsilon_{v,*}} \\ \mathrm{P}_2\left(z_{u,m}|.\right) &\propto \mathrm{p}\left(z_{u,m} = t_k, w_{u,m} = v|\boldsymbol{Z}_{\neg(\boldsymbol{u},\boldsymbol{m})}, \boldsymbol{W}_{\neg(\boldsymbol{u},\boldsymbol{m})}, \boldsymbol{\alpha^{(u)}}, \boldsymbol{\beta^{(k)}}, \boldsymbol{\sigma_k}, \boldsymbol{\tau_v}\right) \end{aligned} \tag{9}$$

$$\begin{aligned} &\propto \left(\frac{c^{-(u,m)}_{k,u,*,1} + \alpha^{(u)}_k}{c^{-(u,m)}_{*,u,*,1} + \alpha^{(u)}_*} + \lambda_1 \frac{c_{k,*,*,1} + \sigma_k}{c_{*,*,*,1} + \sigma_*}\right) \cdot \\ &\left(\frac{c^{-(u,m)}_{k,*,v,1} + \beta^{(k)}_v}{c^{-(u,m)}_{k,*,*,1} + \beta^{(k)}_*} + \lambda_2 \frac{c_{*,*,v,1} + \tau_v}{c_{*,*,*,1} + \tau_*}\right) \end{aligned} \tag{10}$$

The two latent variables are inferred simultaneously in every Gibbs sampling iteration. The topic-labeling process is performed only when $st = 1$. Ultimately, the user-topic distribution ($\boldsymbol{\theta^{(u)}}$) and topic-user distribution ($\boldsymbol{\phi^{(k)}}$) can also be estimated based on Eqs. (11) and (12):

$$\widehat{\theta^{(u)}_k} = \frac{c_{k,u,*,1} + \alpha^{(u)}_k}{c_{*,u,*,1} + \alpha^{(u)}_*} \tag{11}$$

$$\widehat{\phi^{(k)}_v} = \frac{c_{k,*,v,1} + \beta^{(k)}_v}{c_{k,*,*,1} + \beta^{(k)}_*} \tag{12}$$

5.4 Ranking Tags of Non-famous Users

Based on the user-topic distribution $\boldsymbol{\theta^{(u)}}$ of user u obtained through model *Bi-Labeled LDA* (*Bi-Labeled LDA1* or *Bi-Labeled LDA2*), we know the probability user u is interested

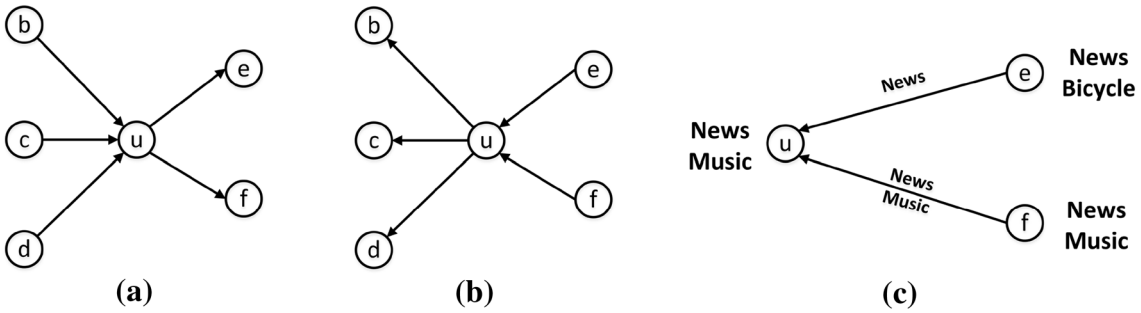

Fig. 3 **a** The following relationship, **b** the interests' spreading, and **c** an example of topic sensitive Random Walk

in each topic. Then, we can rank these topics according to their probabilities and regard tags representing the top topics as u's tag list. This rank sounds reasonable. But for some topics, if only a small number of famous users are interested in them, they may be ranked low in non-famous user's tag lists. For example, though a non-famous user u is quite interested in "lista," u just follows a few famous users who have tag "lista," as this tag is rare among the famous users. This may result in low ranking of "lista" in the tag list of u. On the other hand, if u is really interested in "lista," among the users u follows, the number of non-famous users having this tag may be relatively large. For example, user u may follow a lot of colleagues, who share the same interests such as "lista" with u, but they are not famous users. Based on this observation, we propose to use topic sensitive Random-Walk [19] model to re-rank the tags obtained from *Bi-Labeled LDA* based on following relationship among non-famous users.

Following relationship among non-famous users can be represented by a graph $G(U, E)$ (we call it social network) as illustrated through a toy example in Fig. 3a. An edge of E between nodes in this graph represents a follow relationship between the users denoted by nodes in U. For example, in Fig. 3a, non-famous users b, c, and d follow u, and u follows non-famous users e and f. For a non-famous user u, let $FN(u)$ be the set of all the non-famous users followed by u and $FD(u)$ be the set of all the non-famous users who follows u. For a tag t of user u, we think the more users in $FN(u)$ have this tag, the more user u is interested in the topic represented by tag t, or we simply say the more important tag t is to user u. In other words, we think tags of each following of user u have influence on the importance of each tag to u. To model the influence spreading process, we use Random Walk model, which is illustrated in Fig. 3b. As can be seen from this figure, the topic influence spreading direction is just the opposite to the following relationship shown in Fig. 3a. Tags of users e and f influence the importance of user u's tags, and user u's tags further influence those of users b, c, and d.

Suppose each user u has an initial topic distribution, $\boldsymbol{I}_{\boldsymbol{u}}^{0}$ (equal to $\boldsymbol{\theta}^{(\boldsymbol{u})}$, output of *Bi-Labeled LDA*), representing the initial importance of each tag to the user. This distribution is updated iteratively through influence spreading. Let $I_u^x = (I_{u1}^x, I_{u2}^x, \ldots I_{u|K|}^x)$, denoting user u's topic distribution after the xth iteration, which is updated according to the following equations:

$$I_{uk}^{x+1} = I_{uk}^{x} + \rho \times \sum_{f \in FN(u)} \left(p_{fuk} \times I_{fk}^{x} \right) \tag{13}$$

Table 4 Symbols used in Random Walk model

Symbol	Meaning
$I_u^{(x)}$	Non-famous user u's topic distribution after the xth iteration in the process of Random Walk
$FN(u)$	The set of all the non-famous users followed by u
$FD(f)$	The set of non-famous users who follow user f
ρ	The decay factor in the process of interests spreading
p_{fuk}	The weight of influence of topic t_k spreading from user f to user u

Table 5 Statistics of dataset

Items	Value
# of non-famous users	26,478
# of famous users	14,147
# of follow relationships	2,771,580
# of tweet vocabulary	23,385
# of tags	159

The follow relationships only include the cases that a non-famous user follows a famous user

$$p_{fuk} = \begin{cases} \frac{1}{\sum_{j \in FD(f)} [\theta_k^{(j)} > 0]} & \text{if } \theta_k^{(u)} > 0 \text{ and } \theta_k^{(f)} > 0 \\ 0 & \text{else} \end{cases} \quad (14)$$

where $\rho \in [0, 1]$ is a decay factor, $[\theta_k^{(j)} > 0]$ is an indicator function, $[\theta_k^{(j)} > 0] = 1$, if $\theta_k^{(j)} > 0$, otherwise 0. Equation (13) means that topics t_k can be spread from a following user f to u, if and only if in the results of *Bi-Labeled LDA* both u and f have interest in topic t_k. For topics user u is not interested in, u's following users don't contribute to their importance. The influence of f to each possible follower user u is uniformly. Symbols used in this model are summarized in Table 4.

Figure 3c illustrates the influence spreading process between u's followings and u. Suppose users u, e, and f all have two topics with probability greater than threshold $min\theta$. Users u and f both have tags "news" and "music," and user e has tags "news" and "bicycle." Thus, in the process of Random Walk, tags of "news" and "music" can be transferred from f to u, but only tag "news" can be transferred from e to u and tag "bicycle" cannot be transferred, as u does not have tag "bicycle."

The major steps of random walk model are shown below.

Algorithm: Random Walk
Input: user-topic distribution $\boldsymbol{\theta}$, social network $G(U, E)$, decay factor ρ, threshold $min\theta$ and maximum iteration R
Output: ranked list of tags for each non-famous user
Major steps:
1 for each user $u \in U$
2 Get $\theta^{(u)}$ of every non-famous user u
3 Set $\boldsymbol{I_u^0} = \boldsymbol{\theta^{(u)}}$
4 for $t = 1, \ldots, R$ do
5 for each user $u \in U$
6 for each topic t_k such that $\theta_k^{(u)} > min\theta$
7 $I_{uk}^t \leftarrow I_{uk}^{t-1}$
8 for each user $f \in FN(u)$ with $\theta_k^{(f)} > min\theta$
9 $I_{uk}^t \leftarrow I_{uk}^t + \rho \times p_{fuk}$
10 for each user $u \in U$
11 rank tags based on I_u^R and output top ranked ones with $I_{uk}^R > min\theta$

We call the model combining *Bi-Labeled LDA* with the Random Walk model ***Bi-Labeled-RandomWalk.***

6 Experiments

In this section, we illustrate the efficacy of our proposed methods through an experimental evaluation on real data, comparing with existing state-of-the-art methods. We first show how to extract our experimental dataset and *ground truth*, and then compare the performance of different models proposed in this paper. Finally, we compare them with other existing methods and give a case study.

6.1 Dataset

We use the Twitter graph[6] dataset published by Kwak et al. [6] and the tweets[7] published by Yang et al. [21] as our experimental dataset, which contains a snapshot of the entire Twitter network in 2009 and about 20–30% of all public tweets published on Twitter from June 1, 2009, to December 31, 2009. To make sure that other evaluated methods can get enough text data, we first filter out all the users who post less than 100 tweets during the particular time frame. Given that the original data set is huge, we extract a relatively small network with a *BFS* algorithm. The dataset contains 26,478

[6] http://an.kaist.ac.kr/traces/www2010.html.
[7] http://snap.stanford.edu/data/twitter7.html.

non-famous users, 15,150 famous users, and all their tweets and followings. The detail of our experimental dataset is shown in Table 5.

Using the preprocessing method described in Sect. 4, we get 159 meaningful and qualified tags based on famous users' Twitter List information; 14,147 of the 15,150 famous users have got tags.

To evaluate the performance of the proposed models, we need to compare the inferred interest tags with ***ground truth***, i.e., known interests for some specific Twitter users. To do that, we select those non-famous users who declare their interests in their bios[8] as test dataset. Ding et al. [4] found that users always use "play + NP," "NP fan," "interested in + NP," "love < topic>" or some similar phrases to describe their interests in their biography, where NP stands for a noun phrase. We use the Stanford POS Tagger[9] to find out all the users whose biographies contain such phrases. Finally, we get 3242 such users. Further, we randomly select 120 users from them, and manually tag all the users according to their Twitter homepage, biographies, Lists they created and subscribed to. Note that the reason for manually tagging is that biographies are in free form and ambiguous. For instance, they usually express their interest as someone's fan, such as "Howard Stern fan" and "Orlando Magic fan." As a result, we get 100 users with manually labeled interest tags. Besides, for each selected user, we also classify its interests into several aspects such as {sports(NBA, Orlando Magic, Gator), music, show(Howard Stern)}, where sports is an aspect represented by tags NBA Orlando Magic and Gator. Hence, finally we get 100 users with manually labeled interest tags, which are clustered into several aspects.

6.2 Evaluated Approaches

To evaluate the performance of our proposed models, *Bi-Labeled LDA1*, *Bi-Labeled LDA2 and Bi-Labeled-LDA-RandomWalk*, following models are compared with, and more details about these models are given in Sect. 2.

- *List-Based* This baseline refers to the method proposed by Bhattacharya et al. [1]
- *Labeled LDA-Text* This baseline refers to the approach Labeled LDA [10, 13]. We select the same tag set extracted from famous users' List features as topic labels. It is used to model the generative process of user's tweets and ultimately recommend the words of the tweets to users as their tags.
- *Labeled LDA-Text-Follow* This baseline is the same as *Labeled LDA-Text*, except that it models both the generative process of user's tweets and followings at the same time.
- *Labeled LDA-Follow* This baseline is similar to *Labeled LDA-Text*, except that it models the generative process of user's followings instead of user's tweets, and labels of the top ranked topics instead of tweet words are recommended to users as their final tags.
- *Tag-LDA* This baseline was proposed to model the generative process of words and tags of a labeled document at the same time [15]. Due to the large noise in tweets, we model the generative of hashtags in tweets and famous users' tags at the same time. We finally recommend the hashtags and famous users' tags to users. Different from Labeled LDA, it has no restriction on the topics a document can have.
- *Tag-LDA-Follow* For this baseline, it is the same as *Tag-LDA*, but we replace famous users' tags with users' followings and finally recommend hashtags in tweets to users.

For all the topic models listed above, we set α as 0.5 and β as 0.01. For all the topic models using tweet content listed above, we set the number of topics as 159, the number of tags we extracted for famous users. After learning and inference, we get probability distributions $\theta^{(u)}$ and $\phi^{(k)}$, which indicates a non-famous user u's topic distribution and the topic distribution over terms, respectively. We recommend a term t to user u based on *information gain* measure as used in [7]:

$$p(t|u) = IG(t|u) \propto p(t)\big[p(u|t)\cdot\log p(u|t) + p(\neg u|t)\log p(\neg u|t)\big] + p(\neg t)\begin{bmatrix} p(u|\neg t)\cdot\log p(u|\neg t) \\ +p(\neg u|\neg t)\log p(\neg u|\neg t)\end{bmatrix} \tag{15}$$

Information gain measures the reduction in the entropy associated with user u, incurred by the presence or absence of term t. $\mathrm{p(u)} = {}^{1}/_{|\boldsymbol{U}|}$ is assumed to be the same for all users, and we compute $p(t)$ and $p(u|t)$ as follows:

$$\mathrm{p(t)} = \sum_{u=1}^{|\boldsymbol{U}|}\mathrm{p}(t|u)p(u) = \sum_{u=1}^{|\boldsymbol{U}|}\sum_{k=1}^{|\boldsymbol{K}|}p(u)\cdot p\big(t|z=t_k\big)\cdot p\big(z=t_k|u\big) = \sum_{u=1}^{|\boldsymbol{U}|}\sum_{k=1}^{|\boldsymbol{K}|}p(u)\cdot\phi_t^{(k)}\cdot\theta_k^{(u)} \tag{16}$$

$$\mathrm{p(ult)} = \frac{p(u,t)}{p(t)} = \frac{\sum_{k=1}^{|\boldsymbol{K}|}p(u)\cdot p\big(t|z=t_k\big)\cdot p\big(z=t_k|u\big)}{p(t)} = \frac{\sum_{k=1}^{|\boldsymbol{K}|}p(u)\cdot\phi_t^{(k)}\cdot\theta_k^{(u)}}{p(t)} \tag{17}$$

[8] Bio is a short self-introduction in free form, written by a user in its account profile.

[9] http://nlp.stanford.edu/software/tagger.shtml.

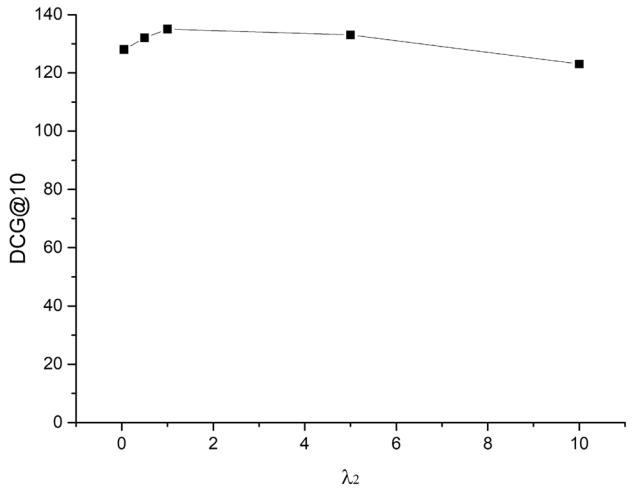

Fig. 4 DCG of *Bi-Labeled LDA2* with different λ_2 and fixed λ_1 (0.05)

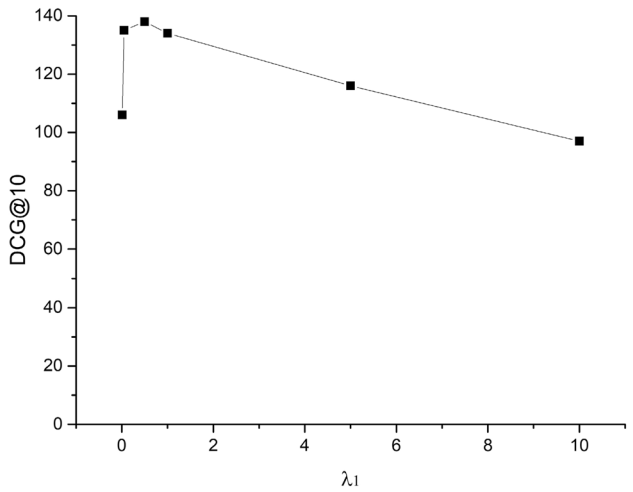

Fig. 5 DCG of *Bi-Labeled LDA2* with different λ_1 and fixed λ_2 (1.0)

After computing the information gain scores, we recommend top scoring terms to users. In addition, we tried several other mechanisms, but this one performs best.

6.3 Comparison of *Bi-Labeled LDA1* with *Bi-Labeled LDA2*

We use two measures to evaluate the performance of each model. One is DCG[10] values of the top n tags extracted for a user by each setting as a measure of performance. The other is the number of the aspects of each user's interests reflected in top n tags ($n \in \{1, 5, 10, 15, 20\}$), which we call *hit number*. Note that, since it is quite difficult to accurately know the exact number of aspects a user is interested in, we use the

[10] http://en.wikipedia.org/wiki/Discounted_cumulative_gain.

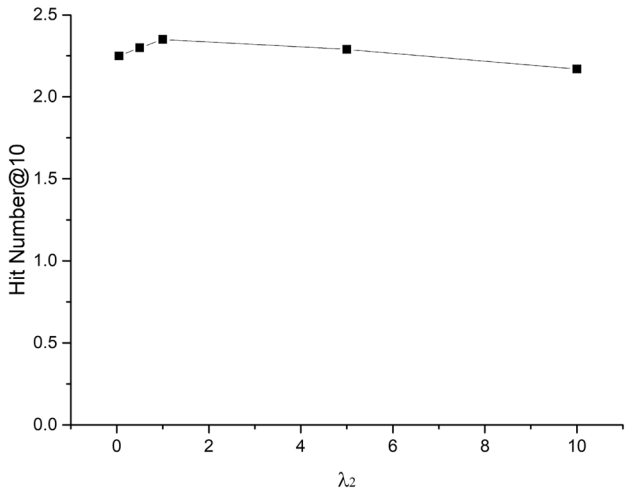

Fig. 6 Hit number of *Bi-Labeled LDA2* with different λ_2 and fixed λ_1 (0.05)

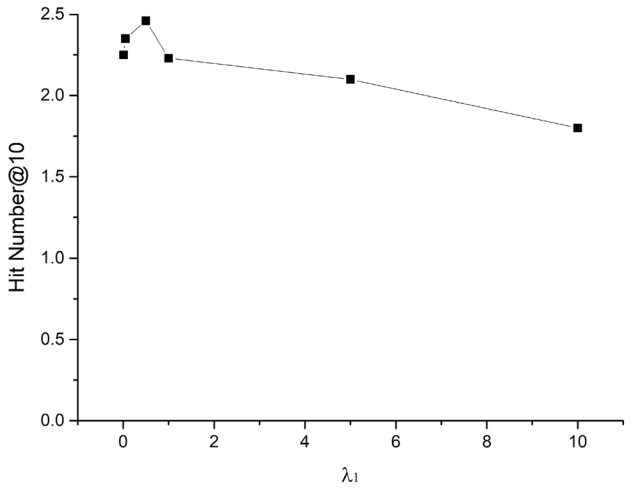

Fig. 7 Hit number of *Bi-Labeled LDA2* with different λ_1 and fixed λ_2 (1.0)

number of interest aspects captured instead of the percentage. In particular, since it is difficult to decide which aspect a user is more interested in, we only consider whether a tag is relevant to a user's interest or not. In addition, even though there are usually more than one tag relevant to one aspect, these tags are usually not completely the same but slightly different. For example, "book, reading, writer, write, kindle" are all relevant to book, but not completely the same. Given that, we calculate DCG and define the graded relevance in Eq. 18 and 19, where tag_i is the tag at rank position i, rel_i is the graded relevance of tag_i, and $tag_i \in k$ means tag_i can reflect u's interest in kth aspect:

$$\text{DCG@n} = \sum_{i=1}^{\text{n}} \frac{2^{rel_i} - 1}{\log_2(i+1)} \tag{18}$$

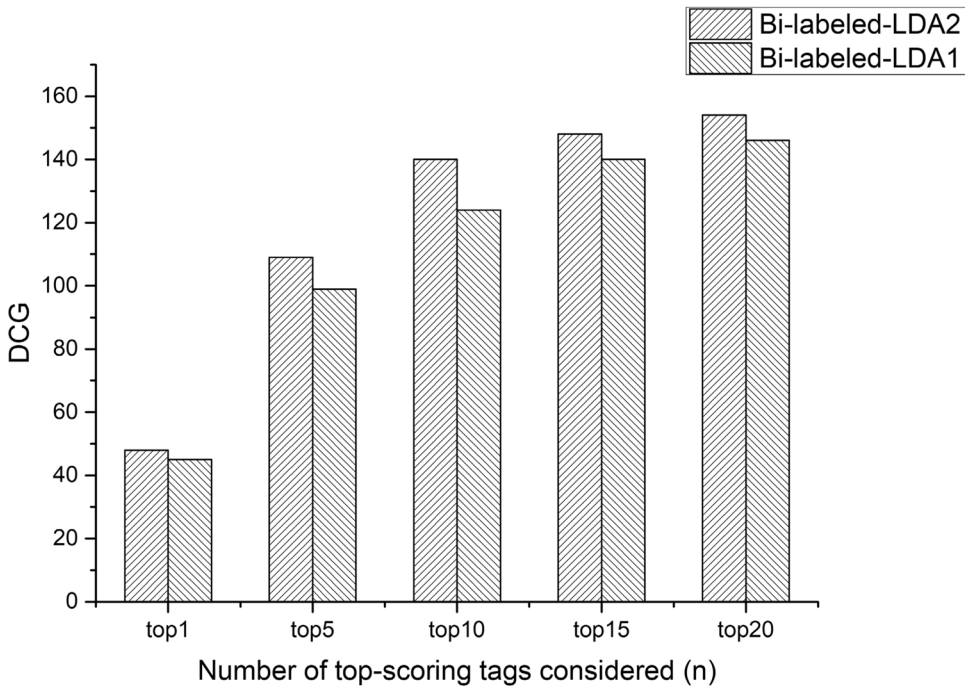

Fig. 8 DCG comparison of *Bi-Labeled LDA1* and *Bi-Labeled LDA2*

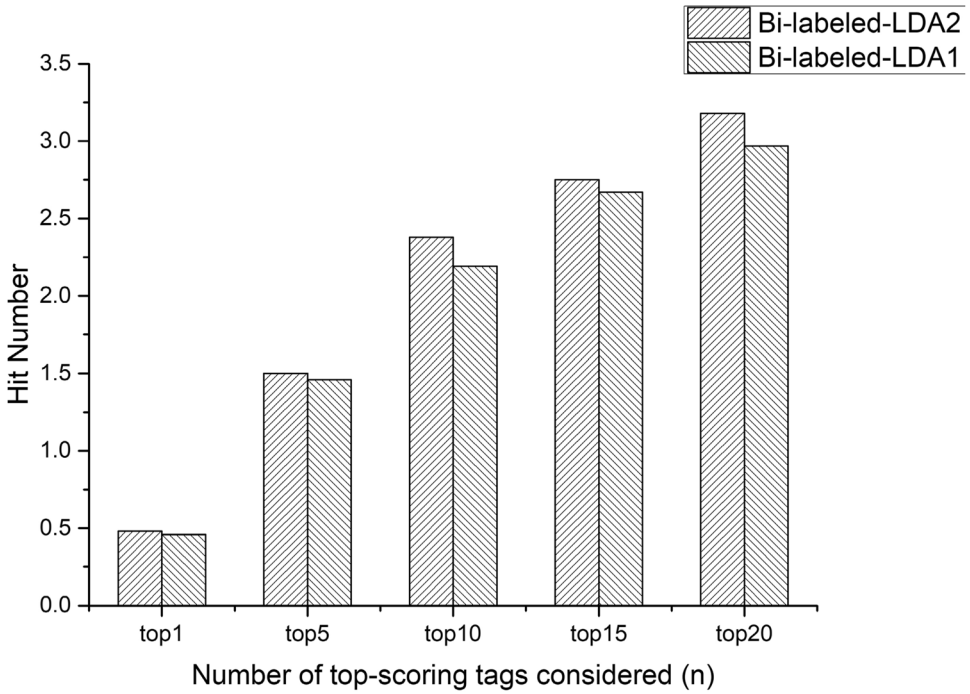

Fig. 9 Hit number comparison between *Bi-Labeled LDA1* and *Bi-Labeled LDA2*

$$\mathrm{rel}_i = \begin{cases} 5 & \text{if } \mathrm{tag}_i \in k, \text{ and } \nexists j < i, \ \mathrm{tag}_j \in k \\ 3 & \text{if } \mathrm{tag}_i \in k, \text{ and } \exists j < i, \ \mathrm{tag}_j \in k \\ 0 & \text{otherwise} \end{cases} \tag{19}$$

In other words, when more than one tag in the tag list of a non-famous user corresponds to a same aspect, the graded relevance of the first one is 5 and the others are 3. In this way, tag sets with top n tags covering all of aspects get the highest score. Specifically, the more aspects a tag list captures and the more tags that reflect different sides of the same aspect, the higher score the tag set will get.

To test effects of parameters λ_1 and λ_2 on the performance of *Bi-Labeled LDA2*, we conducted a set of experiments.

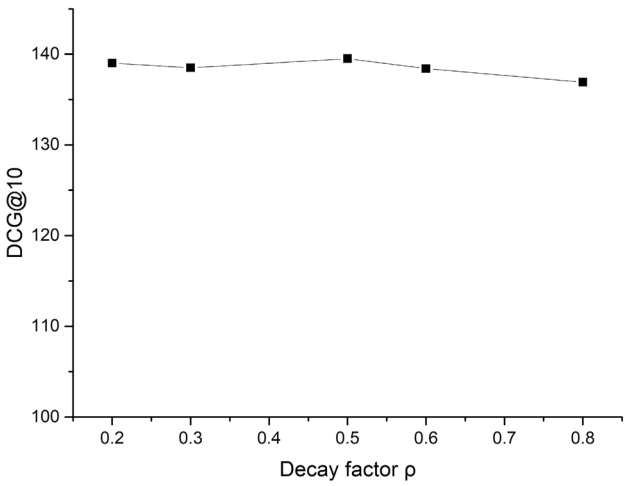

Fig. 10 DCG@10 of *Bi-Labeled-LDA-RandomWalk* with different values of decay factors

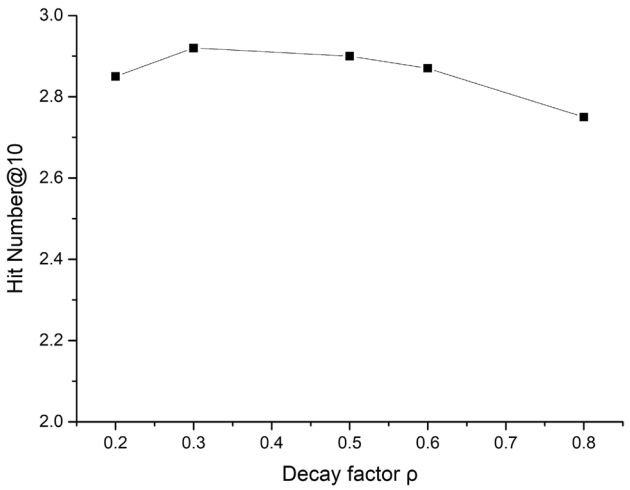

Fig. 11 Hit number of *Bi-Labeled-LDA-RandomWalk* with different values of decay factors

As λ_2 decreases, less and less popularity of famous users is taken into account. And when λ_2 becomes larger, each topic's famous user distribution would become closer and closer to the global distribution of famous users, which is represented as popularity of each famous users but not topic specific. Thus, it leads to the reduction in performance. We fix λ_1 as 0.05 and adjust λ_2 from 0.05 to 10. The measure DCG@10 corresponding to different combinations of λ_1 and λ_2 is shown in Fig. 4. It can be seen from the figure that as λ_2 grows bigger, DCG first increases and then declines.

Lower λ_1 reduces the impact of popular topics and then lowers their rank ordering, but interest topics declared by users usually include some popular topics, such as "sport," "music," "movie," and "travel." We fix λ_2 as 1.0, and let λ_1 vary from 0.01 to 10.0. And the DCG is shown in Fig. 5. Similar to the trend shown in Fig. 4, as λ_1 becomes bigger, the DCG first increases and then decreases.

The hit number of different combinations of λ_1 and λ_2 is shown in Figs. 6 and 7. In Fig. 6, λ_1 is fixed at 0.05, and λ_2 varies from 0.05 to 10, while in Fig. 7, λ_2 is fixed at 1.0, and λ_1 varies from 0.01 to 10. As can be seen from the two figures that too low or too high of λ_1 and λ_2 would result in bad performance. For example, when $\lambda_1 = 0.01$, it is so low that popularity of topics cannot work effectively, and when $\lambda_1 = 10.0$ or $\lambda_2 = 5.0$ or 10.0, they are so large that they can dominate the distribution without distinction among different users and topics. Considering both measures, we finally set the coefficients (λ_1, λ_2) as (0.05,1).

Now, we compare the performance of models *Bi-Labeled LDA1* and *Bi-Labeled LDA2* in Figs. 8 and 9 for the two measures, respectively. Overall, we can see that model *Bi-labeled LDA2* outperforms *Bi-Labeled LDA1*, which means the extension of *Bi-Labeled LDA1* to *Bi-Labeled LDA2* is necessary.

6.4 Comparison of *Bi-Labeled Walk* with *Bi-Labeled LDA2*

To evaluate if Random Walk model is helpful for improving the ranking result, we compare it with *Bi-Labeled LDA2*. In this experiment, we vary decay factor ρ in Eq. 13 from 0.2, 0.3, 0.5, 0.6, to 0.8. The larger the ρ is, the more the influence one can get from its followers. Figures 10 and 11 show DCG@10 and hit number of *Bi-Labeled-LDA-RandomWalk* with different decay factors, respectively.

As can be seen from these figures, the difference between different values of decay factor is not big. Taking into account both DCG and hit number, $\rho = 0.5$ or 0.6 gives relatively better performance. The reason may be that, when ρ is too large, the users would get too much influence from their followers. On the other hand, when ρ is too small, the influence from their followers is too small.

Setting $\rho = 0.5$, we compare *Bi-Labeled-LDA-RandomWalk* with *Bi-Labeled LDA2* in Figs. 12 and 13, which indicates that re-ranking using the random walk model can improve performance.

6.5 Comparing with Existing Methods

Based on ground truth we constructed, we compare our proposed models with state-of-the-art existing models listed in Sect. 5.2. The results are shown in Figs. 14 and 15.

Among the methods based on lists, *Bi-Labeled-LDA1* performs better than *List-Based* which simply ranks the tags through frequency. And *Bi-Labeled-LDA2* which relaxes the assumption of following behavior because of one topic of interest and takes high popularity issues into account outperforms *Bi-Labeled-LDA1*. And re-ranking based on social

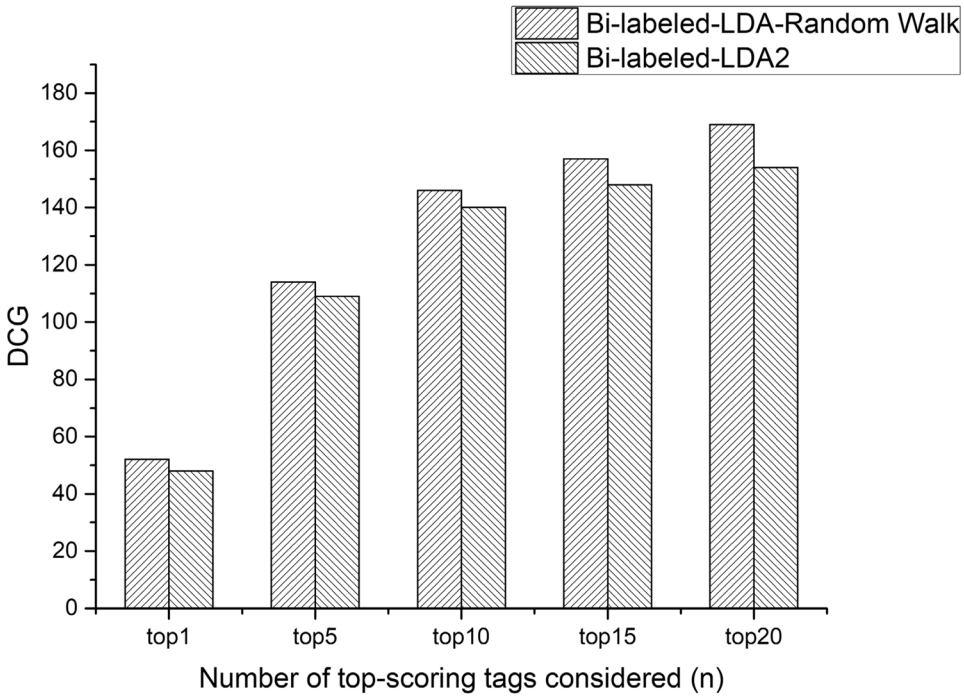

Fig. 12 DCG@10 comparison of *Bi-Labeled-LDA-Random-Walk* with *Bi-Labeled LDA2*

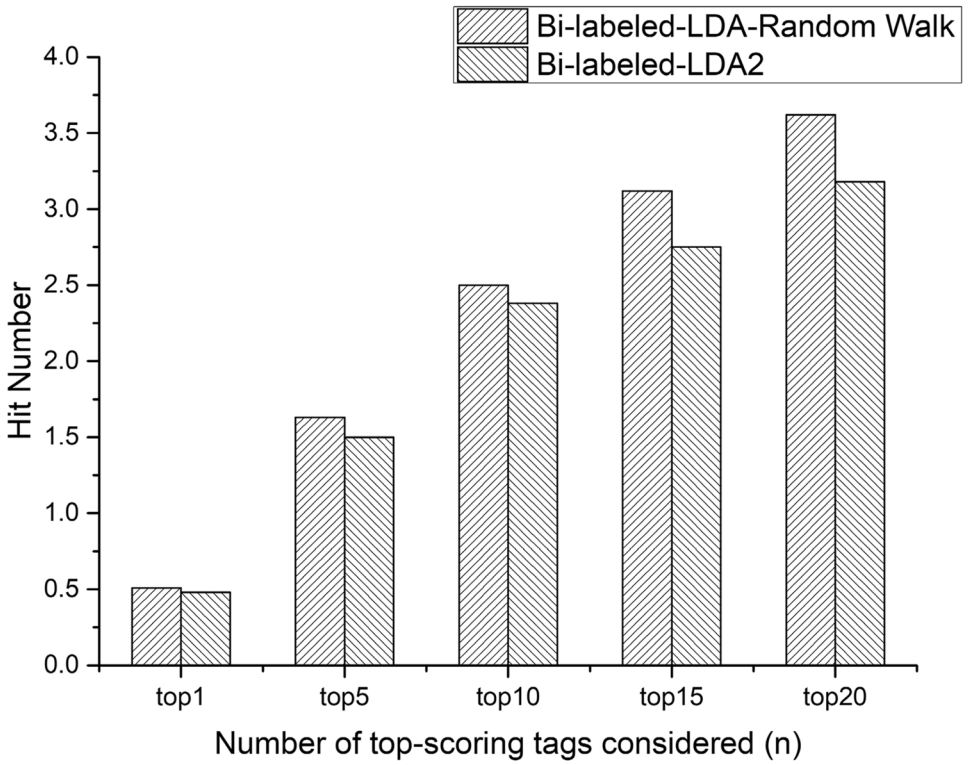

Fig. 13 Hit number comparison of *Bi-Labeled-LDA-Random-Walk* with *Bi-Labeled LDA2*

relationship among normal users is further superior to *Bi-Labeled-LDA2*. We find that tweet-based methods always recommend tags which either relate to daily life, recent events, globally popular topics, or relate to only one or two topics. Even though *List-Based* method can cover many topics of users' interests, it always ranks famous tags such as "news," "movie," "media," and "tech" in the top of the list.

Besides, we actually are very tolerant of the tags recommended by tweet-based method, which are usually not precise enough. For example, tags recommended by them

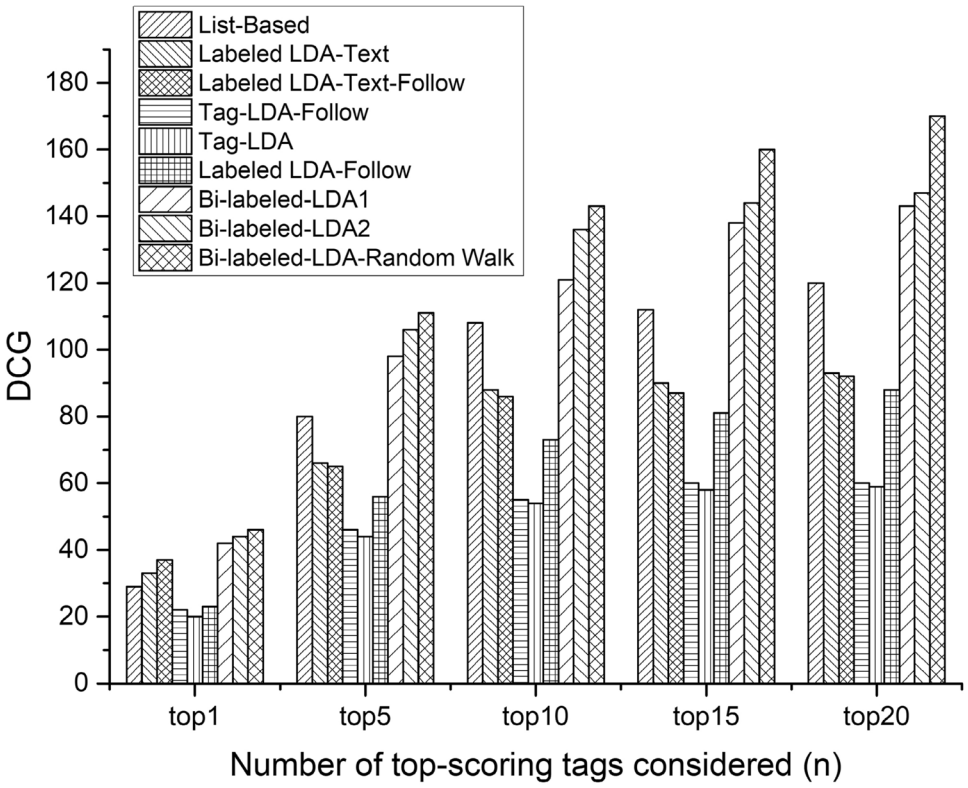

Fig. 14 Comparison of different methods in terms of DCG

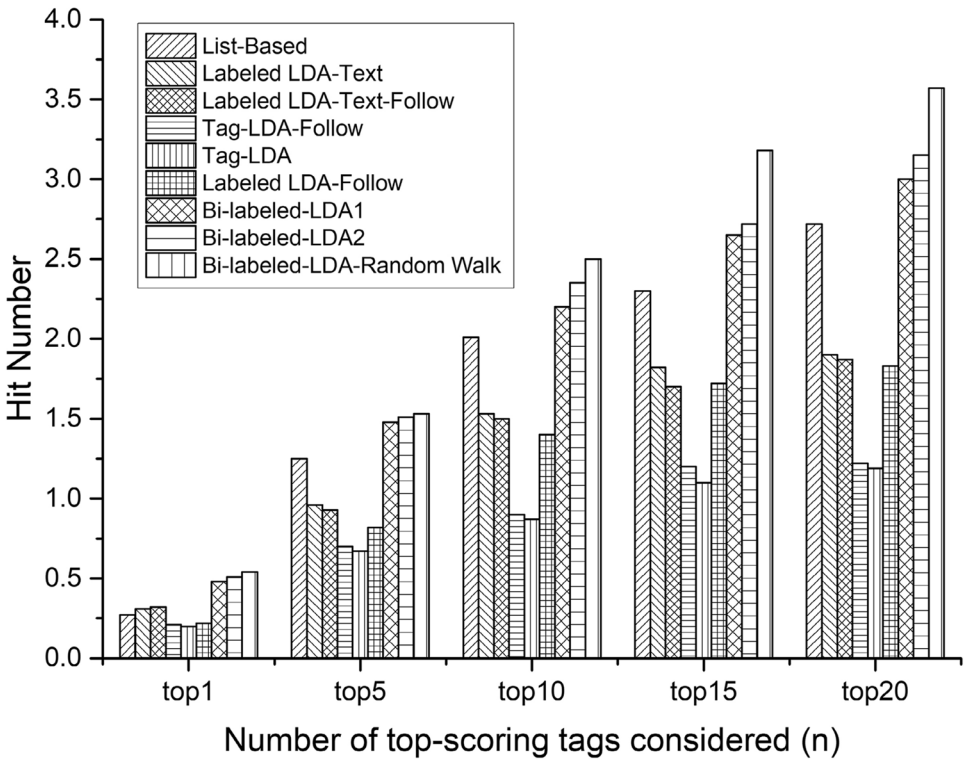

Fig. 15 Comparison of different methods in terms of hit number

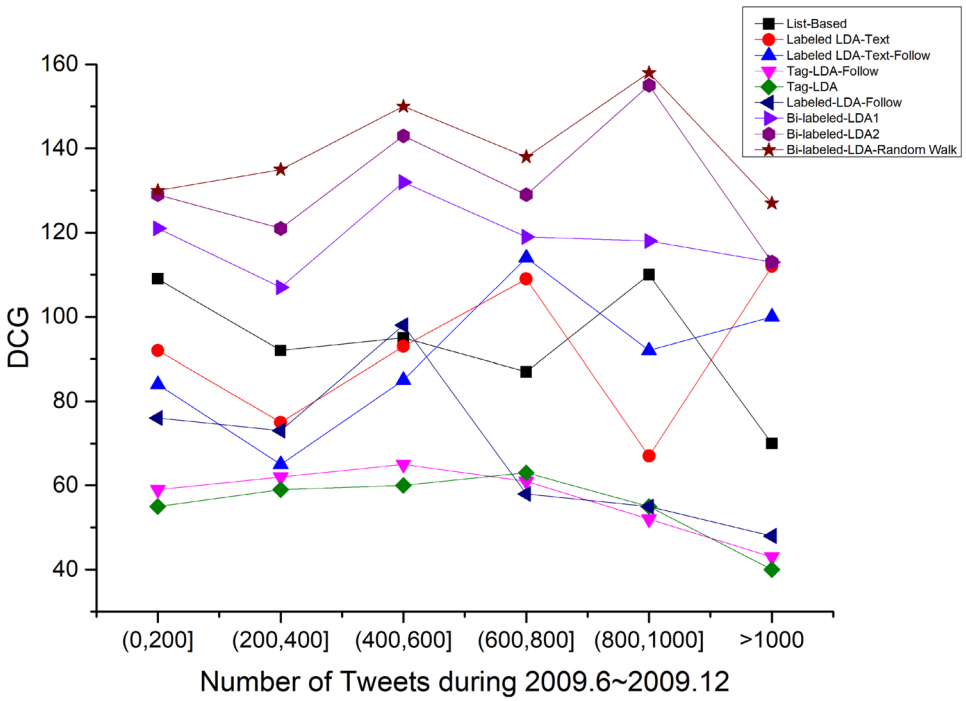

Fig. 16 DCG of each approach and each group of users with a different number of tweets posted during 2009.6–2009.12

for users who are interested in "politics" are usually "government" or "iranelection." Tags recommended by them for users who are interested in "baseball" may be "redsox" (an American professional baseball team), or even "sox." Even though these tags recommended by tweet-based methods are not generalized well enough for other applications, we still treat them as related to users' interests in order to avoid the deviation or artificial evaluation. In this sense, the tags recommended by our method have better generalization and more applicable for many applications, which are further illustrated in Sect. 6.6.

6.5.1 Comparison of Users with Different Levels of Activeness

Finally, in order to evaluate how each model performs for users of different activeness (number of tweets), we adopt the same evaluation method presented in [7] to evaluate the quality of the tag sets extracted by different approaches. We separate the non-famous users into 6 groups with a different number of tweets, including intervals (0,200], (200, 400], (400, 600], (600, 800], (800, 1000], and > 1000, posted from June 1, 2009, to December 31, 2009. We then randomly select 40 users in each group and show them to two annotators, along with the tag sets extracted by each approach without model name. Each tag set contains the top 10 tags recommended by corresponding approach. For each user, the annotators were asked to pick the approach with the best tag set, and they could also pick multiple winners or no winners at all. Then, we report the average DCG for each approach. We also compute the average Kappa statistic of agreement between each pair of annotators on the wins for each approach. The value was 0.85, which signifies a robust agreement between annotators. The DCG of each approach is shown in Fig. 16. As can be seen, our approach consistently outperforms all others for all six groups of users. Among the others, as the number of published tweets increases, models utilizing text information and hashtags first increase and then decrease, which imply both too few and too many tweets are not good. This is easy to understand. Small number of tweets cannot provide enough information, and large number of tweets may provide much noise information.

6.6 Case Study

We illustrate the effectiveness of our proposed models through some cases. Table 6 shows the declared interests for some non-famous users (as given in their bio) and the top 10 tags recommended by the five different methods *List-Based*, *Labeled LDA-Text*, *Tag-LDA-Follow*, *Labeled LDA-Follow,* and *Bi-Labeled-LDA-RandomWalk*. They show relatively better performance in each group of models.

List-Based and *Labeled LDA-Text* perform better compared with other baselines. *Tag-LDA-Follow* performs better then *Tag-LDA*. And *Labeled LDA-Follow* performs well among the baselines which use social network. The tags in bold score 5 in terms of relevance, those in italic score 3, and all the others score 0. It is obvious that the tag sets recommended by *Bi-Labeled LDA-RandomWalk* are more precise and capture a larger fraction of users' interests declared in

Table 6 Top 10 tags and bios of some non-famous users

User with their bio	*List-Based*	*Labeled LDA-Text*	*Tag-LDA-Follow*	*Labeled LDA-Follow*	*Bi-Labeled-LDA-RandomWalk*
wisesumo @CodeSling founder, husband, father of two girls, iOS developer & Jesus follower. My Mantra: Keep Moving Forward	news, media, **tech**, business, social media, marketing, web, world, stuff, influencer	nba, sport, basketball, nfl, book, **tech**, online, **developer**, social media, woman	team, football, **code**, season, web, **app**, open, interesting, health, nfl	**startup,** content, nfl, internet, site, film, star, resource, writer, pro	**startup**, **development**, social media, **tech**, **guru**, speaker, service, blogger, government, content
keepsloanweird: Geek, father (2 boys), #EagleScout, #Cubmaster (Pack 289, Circle 10), private pilot, former #PFE, and overall #MSFT junkie turned #InfoSecengineer/admin. #HYDR	news, celeb, media, stuff, **tech**, **geek**, entertainment, business, peep, web	news, movie, media, organization, show, **tech**, resource, industry, developer, sport	bed, hours, run, phone, kids, weekend, car, early, apple, office	video, **science**, comedian, space, **deal**, youtuber, musician, **game**, comedy, journalist	**geek**, tv, peep, space, **science**, **game**, film, video, hollywood, shopping
leftonred Native New Yorker fixes Computers, Photographs stuff, Drinks Sake, Beer, Wine, Whisky, Spirits & Tea, Creates Origami, Plays Backgammon & Coaches Ping Pong.	news, media, stuff, celeb, entertainment, business, food, **art**, culture, blogger	event, entertainer, fan, science, pr, food, club, app, comedian, **wine**	**nyc**, york, city, park, brooklyn, street, ave, mayor, jersey, train	**beer**, fashion, *wine*, influencer, **nyc**, foody, author, movie, musician, rock	**beer**, **nyc**, *wine*, movie, **geek**, fm, **art**, social media, food, peep
iheni Sino-hippie, Chinese foodie and kickboxer working on accessible UX, mobile and multimedia currently for the Paciello Group, formally BBC. Tweets are my own etc.	**development**, **media**, news, tech, **stuff**, web, celeb, art, design, peep	tech, **social media, web design**, personality, book, phone, blog, geek, interesting, people, write	**web design,** mobile, **stuff,** app, code, bitpage, support, open, phone	resource, **uk**, love, science, news, world, **developer**, tv, inspiration, influencer	**development**, **uk**, actor, *web*, tech, **radio**, startup, education, **web**, **design**, science
steveklein Professor Emeritus/Journalism at George Mason University. Also teach at the University of Mary Washington. I love the Red Wings. I ride Trek. I play TaylorMade.	news, world, media, business, celeb, **journalist**, **sport**, stuff, politics, startup	**cycling**, athlete, tech, nyc, art, uk, agency, health, nfl, basketball	run, **bike**, running, miles, *ride*, race, team, marathon, ran, training	**cycling**, **player**, film, government, **education**, event, personality, interest, life, culture	**journalist**, **cycling**, *journalism*, **sport**, startup, personality, player, **brand**, **education**, nfl

Table 7 Top-10 topics and bios of some famous users

Famous users with their bios	The top 10 topics
Rainn Wilson I am an actor and a writer and I co-created SoulPancake and my son, Walter	Humor, tv, star, fm, hollywood, culture, movie, film, music, peep
Library of Congress We are the largest library in the world, with millions of books, recordings, photographs, maps and manuscripts in our collections	Book, organization, education, government, stuff, news, media, world, info, tech
Dave McClure Geeks. Entrepreneurs. Startups. The Internet Revolution, Act II	Startup, peep, speaker, news, influencer, web, tech, industry, guru, finance
Danah Boyd Internet scholar, social media researcher, youth advocate | Microsoft Research, Harvard Berkman Center	Education, speaker, influencer, blogger, guru, pr, tech, social media, culture marketing
Felicia Day Actress, New Media Geek, Gamer, Misanthrope. I like to keep my Tweets real and not waste people's time	Folk, family, game, youtuber, video, tv, film, entertainment, peep, media

their bios. For instance, Leftonred is a native New Yorker, who is good at fixing computers, playing ping-pong, loving photograph, sake, beer, wine, whiskey, spirits tea, origami, and backgammon. We infer his interests such as "beer," "nyc," "wine," "geek" and "art," while tags extracted by *List-Based* method, *Labeled LDA-Text*, *Tag-LDA-Follow* and *Labeled LDA-Follow* only contain one or two aspects. Even though *List-Based* method can cover many aspects of users' interests, it always ranks very popular tags in the top of its tag set, such as "news," "movie," "media," and "tech." *Labeled LDA-Text* and *Tag-LDA-Follow* which are based on tweet content usually either capture only one aspect of users' interest or recommend tags related to their daily life, recent events, or globally popular topics (e.g., "marathon" and "fb"). Moreover, tags recommended by tweet-based methods are not generalized enough. For example, "bike" and "ride" could be generalized to "cycling," and "code" could be generalized to "development." In this sense, the tags recommended by *Bi-Labeled-LDA-RandomWalk*, *List-Based* and *Labeled LDA-Follow* methods are more generalized and more applicable for applications such as personalized recommendation and advertising.

Furthermore, to see why a famous user v is followed by non-famous users, we calculate their topic distributions as shown in Eq. 20:

$$P(t|v) = \frac{P(t) \times P(v|t)}{P(v)} \tag{20}$$

where $P(v|t)$ is directly from the output, ϕ, of *Bi-Labeled LDA*.

Table 7 shows bios of some famous users and the top 10 topics (tags) with high value of $P(t|v)$.

For example, Rainn Wilson, an American actor who is famous for his Emmy Award-nominated role in television comedy "The Office," is best known because of topics such as "humor," "tv," "star," "hollywood," and "movie." And for Library of Congress of the USA, people usually follow it because of the topics of "book," "organization," "government," and "education." We can see from the table that most of the topics inferred through our proposed model for famous users are accurate and reasonable.

6.7 Discussion

So far, we presented our proposed models through taking Twitter as an example. *Bi-Labeled LDA* is not limited to Twitter. It can be used to other social networking service platforms such as Sina Weibo and Facebook. Social network platforms such as Twitter do not ask users to tag themselves. Others such as Sina Weibo though provide chance for users to provide tags to describe themselves, many users don't use this chance or tags provided are usually ambiguous, trivial, inadequate or even plain false [7]. Thus, in this case, it is also necessary to infer high-quality tags for most users. Our proposed methods can be used in this kind of social networks, where it is easy to get social relationship between users and it is relatively easy to tag a small set of famous users. For example, in Sina Weibo, the platform itself provides high-quality tags for famous users (called "big V users"). We can make use of these tags and the link between famous users and non-famous user to tag other non-famous users utilizing our proposed model. In this case, we skip the step to infer tag of famous users. In case there are not high-quality tags for famous users, we can use their published text information utilizing existing methods based on tweets to get tags for them first, as they are usually active users in terms of publishing behavior.

7 Conclusion

In this paper, we proposed a probabilistic topic model, *Bi-Labeled LDA*, to infer interest tags for non-famous users based on their social relationship with famous users, without using text content information. In particular, the proposed topic model simulates non-famous user's following behavior

and incorporates topic restrictions to both user's topic distribution and topic's word distribution, based on famous user's tag information. The basic model is further extended to relax assumption and consider high popularity issues. To improve the ranking of tags, the model is finally combined with random walk model, utilizing relationship among non-famous users. Experiments conducted on real Tweet dataset show that the proposed models outperform existing models and can capture more topics of user interests. In future, we would like to study how to use the proposed model in other scenarios and evaluate the effects of tags inferred on applications such as personalize recommendation and online advertising.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant Nos. 71771131, 71272029, 71490724, and U1711262.

References

1. Bhattacharya P, Zafar MB, Ganguly N, Ghosh S, Gummadi KP (2014) Inferring user interests in the Twitter social network. In: Proceedings of the 8th ACM conference on recommender systems. ACM, pp 357–360
2. Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3(3):993–1022
3. Cha Y, Bi B, Hsieh CC et al (2013) Incorporating popularity in topic models for social network analysis. In: Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval. ACM, pp 223–232
4. Ding Y, Jiang J (2014) Extracting interest tags from twitter user biographies. In: Information retrieval technology. Springer, Berlin, pp 268–279
5. Ghosh S, Sharma N, Benevenuto F, Ganguly N, Gummadi K (2012) Cognos: crowdsourcing search for topic experts in microblogs. In: Proceedings of the 35th international ACM SIGIR conference on research and development in information retrieval. ACM, pp 575–590
6. Kwak H, Lee C, Park H, Moon S (2010) What is Twitter, a social network or a news media? In: Proceedings of the 19th international conference on world wide web. ACM, pp 591–600
7. Lappas T, Punera K, Sarlos T (2011) Mining tags using social endorsement networks. In: Proceedings of the 34th international ACM SIGIR conference on research and development in information retrieval. ACM, pp 195–204
8. Lim KH, Datta A (2013) Interest classification of Twitter users using Wikipedia. In: Proceedings of the 9th international symposium on open collaboration. ACM, p 22
9. Michelson M, Macskassy SA (2010) Discovering users' topics of interest on twitter: a first look. In: Proceedings of the fourth workshop on analytics for noisy unstructured text data. ACM, pp 73–80
10. Ottoni R, Las Casas D, Pesce JP, Meira Jr W, Wilson C, Mislove A, Almeida V (2014) Of pins and tweets: investigating how users behave across image-and text-based social networks. AAAI ICWSM
11. Quercia D, Askham H, Crowcroft J (2012) TweetLDA: supervised topic classification and link prediction in Twitter. In: Proceedings of the 4th annual ACM web science conference. ACM, pp 247–250
12. Rakesh V, Singh D, Vinzamuri B, Reddy CK (2014) Personalized recommendation of twitter lists using content and network information. In: Eighth international AAAI conference on weblogs and social media
13. Ramage D, Hall D, Nallapati R, Manning CD (2009) Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of the 2009 conference on empirical methods in natural language processing, vol 1. Association for Computational Linguistics, pp 248–256
14. Resnik P, Hardisty E (2010) Gibbs sampling for the uninitiated (No. CS-TR-4956). Maryland university college park institute for advanced computer studies
15. Si X, Sun M (2009) Tag-LDA for scalable real-time tag recommendation. J Comput Inf Syst 6(1):23–31
16. Steck H (2011) Item popularity and recommendation accuracy. In: Proceedings of the fifth ACM conference on recommender systems, RecSys'11, New York, NY, USA. ACM, pp 125–132
17. Wagner C, Liao V, Pirolli P, Nelson L, Strohmaier M (2012) It's not in their tweets: modeling topical expertise of twitter users. In: Privacy, security, risk and trust (PASSAT), 2012 international conference on social computing (SocialCom). IEEE, pp 91–100
18. Wang T, Liu H, He J, Du X (2013) Mining user interests from information sharing behaviors in social media. In: Advances in knowledge discovery and data mining. Springer, Berlin, pp 85–98
19. Weng J, Lim EP, Jiang J et al (2010) Twitterrank: finding topic-sensitive influential twitterers. In: Proceedings of the third ACM international conference on web search and data mining. ACM, pp 261–270
20. Xu Z, Lu R, Xiang L, Yang Q (2011) Discovering user interest on twitter with a modified author-topic model. In: 2011 IEEE/WIC/ACM international conference on Web intelligence and intelligent agent technology (WI-IAT), vol 1. IEEE, pp 422–429
21. Yang J, Leskovec J (2011) Patterns of temporal variation in online media. In: Proceedings of the fourth ACM international conference on web search and data mining. ACM, pp 177–186
22. Zhao D, Rosson MB (2009) How and why people Twitter: the role that micro-blogging plays in informal communication at work. In: Proceedings of the ACM 2009 international conference on supporting group work. ACM, pp 243–252
23. Zhao X, Jiang J (2011) An empirical comparison of topics in twitter and traditional media. Singapore Management University School of Information Systems Technical paper series. Retrieved November, 10, 2011