Incorporating Popularity in Topic Models for Social Network Analysis

Youngchul Cha Bin Bi Chu-Cheng Hsieh Junghoo Cho
UCLA Computer Science Dept
Los Angeles, CA 90095
{youngcha, bbi, chucheng, cho}@cs.ucla.edu

ABSTRACT

Topic models are used to group words in a text dataset into a set of relevant topics. Unfortunately, when a few words frequently appear in a dataset, the topic groups identified by topic models become noisy because these frequent words repeatedly appear in "irrelevant" topic groups. This noise has not been a serious problem in a text dataset because the frequent words (e.g., *the* and *is*) do not have much meaning and have been simply removed before a topic model analysis. However, in a social network dataset we are interested in, they correspond to popular persons (e.g., *Barack Obama* and *Justin Bieber*) and cannot be simply removed because most people are interested in them.

To solve this "popularity problem", we explicitly model the popularity of nodes (words) in topic models. For this purpose, we first introduce a notion of a "popularity component" and propose topic model extensions that effectively accommodate the popularity component. We evaluate the effectiveness of our models with a real-world Twitter dataset. Our proposed models achieve significantly lower perplexity (i.e., better prediction power) compared to the state-of-the-art baselines.

In addition to the popularity problem caused by the nodes with high incoming edge degree, we also investigate the effect of the outgoing edge degree with another topic model extensions. We show that considering outgoing edge degree does not help much in achieving lower perplexity.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*clustering*; D.2.8 [**Software Engineering**]: Metrics—*performance measures*

General Terms

Experimentation, Algorithms

SIGIR'13, July 28–August 1, 2013, Dublin, Ireland.

Keywords

topic model, social-network analysis, popularity bias, handling popular users, Latent Dirichlet Allocation

1. INTRODUCTION

Microblogging services such as Twitter are popular these days because they empower users to broadcast and exchange information or thoughts in realtime. Distinct from other social network services, relationships on Twitter are unidirectional and often interest-oriented. A user may indicate her interest in another user by "following" her, and previous studies [16, 23] show that users are more likely to follow people who share common interests, even though "following relationships" among users look unorganized and haphazard at first glance. Thus, if we can correctly identify the shared hidden interests behind users' following relationships, we can recommend more relevant users and group users sharing common interests in the social network services.

In this paper, we refine topic models to correctly identify the hidden interests behind users' following relations (instead of their tweets as in [30]). A topic model is a statistical model originally developed for discovering hidden topics from a collection of documents. It postulates that every document is a mixture of topics, and words in a document are attributable to these hidden topics. Here, we posit that the following relations are not random but are interest-attributable. Then, we can discover the hidden interest behind each following relation by regarding a user's following list as a document, and each person in the user's following list as a word. Now, topic models can easily help us correctly identify the hidden interests and derive a low dimensional representation of the observed following lists.

However, simply applying topic models to the follow-relation analysis may cause some problems. Our previous study [7] reported significant clustering quality degradation when the authors directly applied Latent Dirichlet Allocation (LDA) [5] to Twitter's following relation dataset. As LDA is built on an assumption that every word in a document should be of roughly equal popularity, stop words like *the* and *is* must be removed in preprocessing stages. However, keeping popular persons like *Barack Obama* and *Justin Bieber* in a user's following list can be beneficial for the following reasons: (1) These well-known users can work as an effective labels of identified topic groups. For example, when a group contains well-known politicians like Barack Obama, we may immediately identify that the group is likely to be on politics.[1] (2) Many new Twitter users do not know who is on Twitter and who is not, so they often fail to follow popular users of potential interest, not knowing their presence. If the system can recommend interesting and well-known users of interest, it can significantly improve the users' return rate and stickiness [31].

In this work, we propose refined topic models specialized in handling the quality degradation caused by a limited number of popular users, which we call "popularity bias". For this, we introduce a notion of a "popularity component" and explore various ways to effectively incorporate it. We also evaluate the effectiveness of our proposed topic models with the widely used "perplexity" [2] calculated over the real-world Twitter following relation dataset.

Note that the popularity bias is not limited to social network datasets. This bias often appears in datasets showing user's preference over some popular items (or nodes) such as webpage visit logs, advertisement click logs, product purchase logs, etc. We believe our proposed models can effectively improve the quality of recommendations and clusterings in the web services generating such logs.

In summary, we make the following contributions in this paper:

- We propose topic models appropriate for social-network analysis. We introduce a popularity component, which explicitly models popularity of users, and explore various ways to incorporate it step by step.

- We conduct extensive experiments using a real-world Twitter dataset. Through these experiments, we demonstrate that our models are very effective in recommending more relevant users with significantly lower perplexity than the state-of-the-art baselines.

- We show that there is no clear relationship between how many persons a user follows (i.e., outgoing edge degree) and how topic sensitive she is. It is quite different from the popularity bias which shows that if a user is followed many times (i.e., high incoming edge degree), many users follow her because she is just popular.

2. RELATED WORK

In this section, we briefly review topic models related to social-network analysis in three categories: topic models for authorship, hypertexts, and edges.

The topic models in the first category were proposed to analyze documents (texts) with their authors. As these models incorporate authors and their relationships in the model, they can be viewed as early forms of social-network topic models. They attempt to group documents and authors by assuming that a document is created by authors sharing common topics. The concept of authors (users) was initially introduced by Steyvers et al. [34] in the Author-Topic (AT) model. With the additional co-author information, they could successfully extract hidden research topics and trends from CiteSeer's abstract corpus. The AT model was extended to the the Community User Topic (CUT) model by Zhou et al. [41] to capture semantic communities. McCalum et al. [24] also extended the AT model and proposed the Author-Recipient-Topic (ART) model and the Role-Author-Recipient-Topic (RART) model to analyze e-mail networks. Pathak et al. [29] modified the ART model and suggested the Community-Author-Recipient-Topic (CART) model, which is similar to the RART model. In addition to these AT model family, other LDA extensions and probabilistic topic models were also proposed to analyze chat data [36], voting data [38], annotation data [22], and tagging data [17].

The topic models in the second category are more closely related to social-network analysis and analyze documents with their citations (i.e., hypertexts). Cohn et al. [10] initially introduced a topic model combining PLSI [20] and PHITS [9]. Later, PLSI in this model was replaced with LDA [5] by Erosheva et al. [13]. Nallapati et al. [27] extended Erosheva's model and proposed Link-PLSA-LDA model which applies PLSI and LDA to cited and citing documents, respectively. Chang et al. [8] also proposed the Relational Topic Model (RTM) which models a citation as a binary random variable. Dietz et al. [12] proposed a topic model to analyze topical influence of research publications. More sophisticated models were proposed by Gruber et al. [15] and Sun et al. [35]. Hybrid approaches were also attempted. Mei at al. [25] introduced a regularized topic modeling framework incorporating a graph structure and Nallapati et al. [26] combined network flow and topic modeling.

The topic models in the last category only uses linkage (edge) information. Since they only focus on the graph structure, they can be easily applied to a variety of datasets. However, there has been relatively less research in this category. Our work belongs to this category and focuses on solving the issue caused by popular nodes in the graph structure. Airoldi et al. [3] proposed the Mixed Membership Stochastic Block (MBB) model to analyze pairwise measurements such as social networks and protein interaction networks. Zhang et al. [40] and Henderson et al. [18] dealt with the issues in applying LDA to academic social networks. The former focused on the issue of converting the co-authorship information into a graph, and proposed edge weighting schemes based on collaboration frequency. The latter addressed the issue of a large number of topic clusters generated due to "low popularity" nodes in the network. The "high popularity" issue was initially addressed by Steck [32]. He defined a new metric called Popularity-Stratified Recall and suggested a matrix factorization method optimizing it. In our previous study [7], we investigated the issue in more detail and proposed effective solutions based on probabilistic topic models. However, our previous solutions are rather heuristic and more about how to tune topic models to handle the high popularity issue. In this paper, we take a more principled approach to this issue and propose more effective topic models.

3. FOLLOW-RELATION ANALYSIS USING LDA

In this section, we explain how to apply topic models to social network analysis. After briefly explaining two probabilistic topic models, Probabilistic Latent Semantic Indexing (PLSI) and Latent Dirichlet Allocation (LDA), we introduce

[1]Existing topic models simply identify a group of words (or users) that belong to a topic group, not the semantic labels of each group.

[2]It measures prediction power of a trained model. The definition is given in Section 5.2

the challenge of a "popularity bias" which we address in this study.

3.1 Topic Models

Topic models are built on the assumption that there are latent variable(s) behind each observation in a dataset. In the case of a document corpus, the usual assumption is that there is a hidden topic behind each word. PLSI [20] introduced a probabilistic generative model to Latent Semantic Indexing (LSI) [11], one of the most popular topic models. Equation (1) represents its document generation process based on the probabilistic generative model:

$$p(d,w) = p(d)p(w|d) = p(d)\sum_{z\in Z} p(z|d)p(w|z). \quad (1)$$

$p(d,w)$ denotes the probability of observing a word w in a document d and can be decomposed into two parts: $p(d)$, the probability distribution of documents, and $p(w|d)$, the probability distribution of words given a document. This equation describes a word selection process for a document, where an author first selects a document then a word in that document. By repeating this selection process sufficiently, we can generate a full document and eventually a whole document corpus. Based on the assumption that there is a latent topic z for each word w, the equation above can be rewritten with the multiplication of $p(w|z)$, the probability distribution of words given a topic, and $p(z|d)$, the probability distribution of topics given a document. In this way, an additional topic selection step is added between the document selection step and the word selection step. We sum the multiplication over a set of all independent topics Z because there exist a number of possible topics from which a word may be derived.

The goal of the topic model analysis is to accurately infer $p(w|z)$ and $p(z|d)$. Given the probabilistic generative model explained above, we can effectively infer $p(w|z)$ and $p(z|d)$ by maximizing the log-likelihood function L of observing the entire corpus as in Equation (2):

$$\begin{aligned} L &= \log[\prod_{d\in D}\prod_{w\in W} p(d,w)^{n(d,w)}] \\ &= \sum_{d\in D}\sum_{w\in W} n(d,w)\log p(d,w), \quad (2) \end{aligned}$$

where $n(d,w)$ denotes the word frequency in a document. The inferred $p(w|z)$ and $p(z|d)$ measure the strength of association between a word w and a topic z and that between a topic z and a document d, respectively. For example, if $p(w_{vehicle}|z_{car}) > p(w_{technology}|z_{car})$, the word *vehicle* is more closely related to the topic *car* than the word *technology*, though they are all related to the topic *car*. In this way, PLSI and other probabilistic topic models support multiple memberships and produce more reasonable clustering results.

Although PLSI introduced a sound probabilistic generative model, it showed a poor performance when predicting unobserved words and documents. To solve this "overfitting" problem, LDA [5] introduced Dirichlet priors α and β to PLSI, to constrain $p(z|d)$ and $p(w|z)$, respectively. α is a vector of dimension $|Z|$, the number of topics, and each element in α is a prior for a corresponding element in $p(z|d)$. Thus, a higher α_i implies that there are more frequent prior observations of topic z_i in a corpus. Similarly, β is a vector of dimension $|W|$, the number of words, and each element in β is a prior for a corresponding element in $p(w|z)$. By placing Dirichlet priors α and β on the multinomial distributions $p(z|d)$ and $p(w|z)$, these posterior distributions are smoothed by the amount of priors α and β, and the model becomes safe from PLSI's overfitting problem. As a conjugate prior for the multinomial distribution, the Dirichlet distribution also simplifies the statistical inference and enables the use of the collapsed Gibbs sampling [33]. It is also known that PLSI emerges as a specific instance of LDA under Dirichlet priors [14, 19].

3.2 LDA on Recommending Who to Follow

In a social network service, a user r's following another user w can be intuitively interpreted as the user r (acting as a "reader") expresses her interest in tweets written by the user w (acting as a "writer"). We believe this interest plays a role in the establishment of the following relation (or edge) as the topic does in the document generation process explained in Section 3.1. In this study, we assume that there exists a follow edge generative model: a reader first chooses an interest, and based on the chosen interest, the reader chooses a writer to follow. In this model, a document in a corpus becomes a reader's following list, and a word becomes a writer in the list.

Analyzing Twitter follow edges using LDA delivers two estimates: $p(z|r)$ and $p(w|z)$. $p(z|r)$ indicates a reader r's interest distribution and $p(w|z)$ indicates a writer w's importance in an interest group z. Thus, $p(w|z)$ can be used in clustering Twitter users having the same interest. From Equation (1), we can easily estimate $p(w|r)$, the likelihood of a reader r's following a writer w, which can be used for recommendation.

When we apply topic models to a social network dataset, we notice the following differences [7]:

1. In a document generative model, a word is sampled *with replacement.* However, in our follow edge generative model, a reader cannot follow the same writer twice. Thus, a writers should be sampled *without replacement.*

2. When analyzing a textual dataset, common entries like *the* and *is* are simply ignored because they do not have important meaning. Thus, they are called "stop words". However, in a social network dataset, these entities correspond to "celebrities" like *Barack Obama* and *Justin Bieber* who attract more followers than others. Thus, they cannot be simply ignored but should be carefully handled.

Because of the first difference, some probability distributions in our model follow *multivariate hypergeometric distributions* instead of *multinomial distributions.* This difference is important because LDA benefits from Dirichlet priors, which are conjugate priors of multinomial distributions. However, it is known that a multivariate hypergeometric distribution converges to a multinomial distribution as the population size grows large [1]. Since millions of users are included in our Twitter dataset, we can disregard the consequence caused by the sampling *without replacement.*

The second difference affects the quality of a topic model analysis. When celebrities are simply included without any special handling, they appear even in irrelevant topic groups and make the topic analysis severely biased to them. Such a "popularity bias" can be seen everywhere, from purchase logs to movie rating data. In the next section, we propose refined topic models which address this popularity bias.

4. REFINED TOPIC MODELS

In this section, we introduce a notion of a "popularity component" using a *simple model*, which acts as a base of our later models. We propose three refined models, and discuss how they may ease the popularity bias. At the end of this section, we explore various extensions to these refined models.

4.1 Simple Model

As described in Section 3.2, in our follow edge generative models, a reader first selects an interest (a topic) from a distribution $p(z|r)(\theta)$, and then selects a writer from a distribution $p(w|z)(\phi)$. The θ and ϕ are constrained by Dirichlet priors α and β, respectively. This process is depicted in a plate notation in Figure 1(a). We formulate the probability for a reader a to follow a writer b based on an interest z (or $z_{a,b}$) as follows:

$$p(z_{a,b}|\cdot) = p(z|r_a)p(w_{a,b}|z). \tag{3}$$

Note that this equation is equivalent to Equation (1), and we use $w_{a,b}$ to indicate a follow edge between a reader a to a writer b. By considering the Dirichlet priors α and β, the same probability can be represented in LDA as follows:

$$p(z_{a,b}|\cdot) \propto \int p(z|\theta)p(\theta|\alpha)d\theta \times \int p(w_{a,b}|z,\phi)p(\phi|\beta)d\phi. \tag{4}$$

In a *simple model*, we incorporate a "popularity component" into LDA, as in Figure 1(b). The popularity component (in a dotted box) consists of a multinomial distribution π, which represents an in-corpus writer distribution, and a Dirichlet prior γ constraining π. Note that γ is a vector of length J, the number of unique writers, and each element has a value of $\gamma_w = \frac{f_w}{f_*}$, where f_w denotes an in-corpus frequency of a writer w (i.e., the number of followers to the writer), and f_* denotes a total frequency ($\sum_w f_w$). Thus, in the *simple model*, when a reader follows a writer, the writer selection probability ϕ is multiplied by π so that popular writers are weighted accordingly. We formulate this change (from Equation (4)) into the following equation:

$$p(z_{a,b}|\cdot) \propto \int p(z|\theta)p(\theta|\alpha)d\theta \times \iint p(w_{a,b}|z,\phi,\pi)p(\phi|\beta)p(\pi|\gamma)d\phi d\pi. \tag{5}$$

4.2 Polya-Urn Model

Although the *simple model* incorporates the popularity component in LDA, this incorporation is too simple. Whenever a reader follows a writer, the model favors a popular writer according to her in-corpus distribution π. However, the in-corpus writer distribution can be largely different from a in-topic writer distribution. For example, though *Barack Obama* is more popular than *Justin Bieber* in general, *Justin Bieber* is more popular than *Barack Obama* among people who like music. Thus, the writer should be picked up from the in-corpus distribution or the in-topic distribution. When every topic is assumed to be equally likely (as in LDA's symmetric prior assumption), the in-corpus writer distribution is the sum of per-topic (in-topic) writer distributions ($p(w) = \sum_z p(w|z)p(z)$), and we may consider the former as a global writer distribution and the latter as a local writer distribution. Since a writer is picked up from her global (in-corpus) distribution or local (per-topic) distribution, we may represent a popularity-incorporated writer distribution as a mixture of the global distribution and the local distribution. This interpretation leads us to a *polya-urn model* depicted in Figure 1(c). In [2], the authors took a similar approach for a topic distribution θ to capture global topics as well as local topics.

Figure 1(c) depicts how the global and local distribution are populated with the popularity component depicted in the dotted box. In addition to the γ and π in the *simple model*, the popularity component in the *polya-urn model* has a concentration scalar λ. Initially, the multinomial distribution π is generated from the Dirichlet prior γ. Then, π works as a Dirichlet prior for ϕ, together with the concentration scalar λ. As λ works as a weight to the prior observation π, ϕ becomes similar to π when λ has a high value. On the other hand, ϕ deviates from π when λ has a low value. Since π works as a base distribution and ϕ deviates from π per topic, π can be considered as a global (in-corpus) writer distribution, and ϕ can be considered as a local (per-topic) writer distribution.

To derive a collapsed Gibbs sampling equation for the *polya-urn model*, we define $c_{k,m,j}$ as the number of associations between a topic z_k and a writer w_j followed by a reader r_m (or a follow edge from a reader r_m to a writer w_j) as in Equation (6) [33]:

$$c_{k,m,j} = \sum_{n=1}^{N_m} I(z_{m,n} = k \& w_{m,n} = j). \tag{6}$$

We also define $c_{k,m,j}^{-(a,b)}$ as the count when we exclude the edge from a reader r_a to a writer w_b. Then the collapsed Gibbs sampling equation of LDA (derived from Equation (4)) becomes:

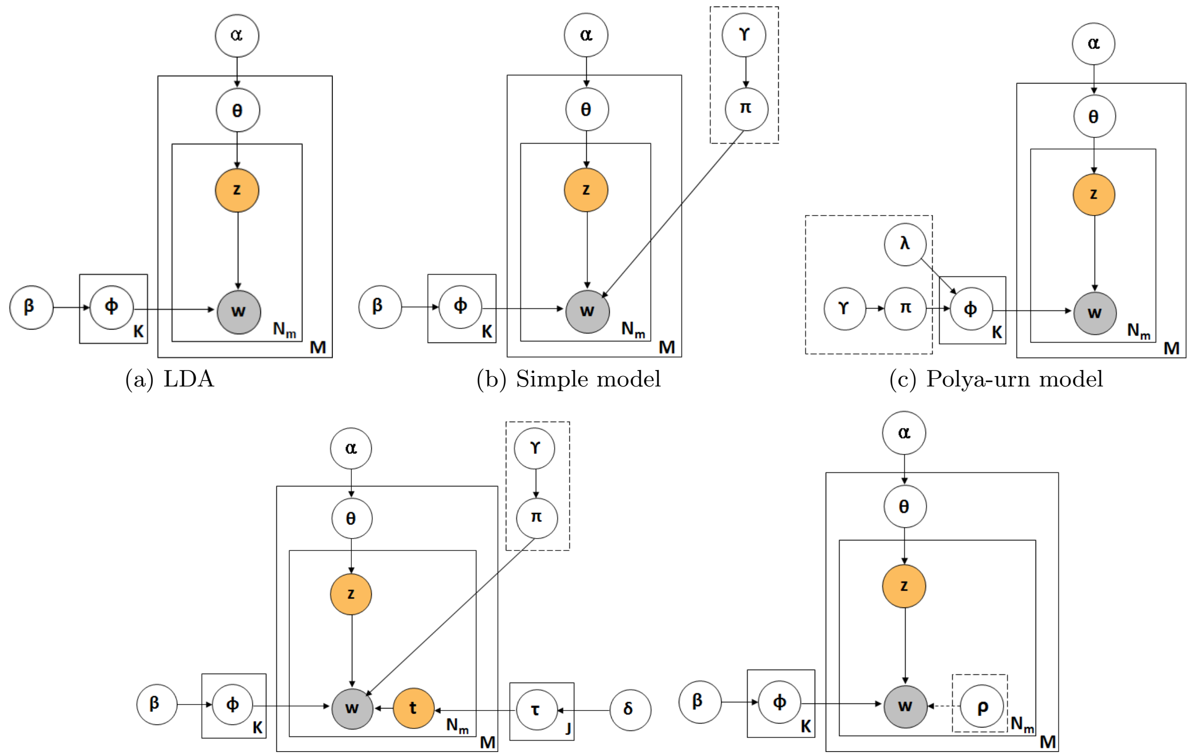
$$p(z_{a,b}|\cdot) \propto \frac{c_{z_{a,b},a,*}^{-(a,b)} + \alpha_{z_{a,b}}}{c_{*,a,*}^{-(a,b)} + \alpha_*} \times \frac{c_{z_{a,b},*,w_{a,b}}^{-(a,b)} + \beta_{w_{a,b}}}{c_{z_{a,b},*,*}^{-(a,b)} + \beta_*}, \tag{7}$$

where the symbol * denotes a summation over all possible subscript variables. As we select a writer from a mixture of a global and a local writer distribution, the topic assignment probability of the *polya-urn model* should be extended to:

$$p(z_{a,b}|\cdot) \propto \frac{c_{z_{a,b},a,*}^{-(a,b)} + \alpha_{z_{a,b}}}{c_{*,a,*}^{-(a,b)} + \alpha_*} \times \left(\frac{c_{z_{a,b},*,w_{a,b}}^{-(a,b)}}{c_{z_{a,b},*,*}^{-(a,b)}} + \lambda \frac{c_{*,*,w_{a,b}} + \gamma_{w_{a,b}}}{c_{*,*,*} + \gamma_*}\right). \tag{8}$$

Note that the global distribution dominates in the mixture as the concentration parameter λ increases. On the other hand, as λ decreases, the local distribution dominates and the whole equation becomes similar to that of LDA.

4.3 Two-Path Model



(a) LDA (b) Simple model (c) Polya-urn model

(d) Two-path model (e) Weight model

Figure 1: LDA and proposed topic models

In the *polya-urn model*, when a reader follows a writer, she first selects a topic, and then selects a writer from the mixture of a global and a local writer distribution for the selected topic. Although the mixture explains non-topic related (popularity-based) following relations as well as topic related (interest-based) ones, the initial topic selection process is common in both cases. In a *two-path model*, we clearly separate the non-topic related following relations from the topic related ones by assuming that there are two separate paths from which a writer can come. This separation in early stage is expected to help generate more clear topics. For this separation, we introduce a new binary latent variable t which indicates the path the writer comes from. $t = 1$ means that the writer comes from a "topic path" and $t = 0$ means that she comes from a "popularity path". Now, we do a "path-labeling" as well as a "topic-labeling" for a follow edge, and our goal is to accurately infer t as well as z (when $t = 1$).

Figure 1(d) depicts this *two-path model*. The variable t follows a Bernoulli distribution τ which is constrained by a Beta prior δ. As a more popular writer may have a higher probability of being followed through the popularity path than a less popular writer, we pose an asymmetric prior according to the writer's popularity. For example, larger portion of edges to *Barack Obama* will be labeled with the popularity path because he has lower δ and τ. We extend Equation (6) with the new path indicator variable t:

$$c_{k,m,j,s} = \sum_{n=1}^{N_m} I(z_{m,n} = k \& w_{m,n} = j \& t_{m,n} = s). \tag{9}$$

Then, the path-labeling and the topic-labeling probability are derived as:

$$p(t_{a,b}|\cdot) \propto \frac{c^{-(a,b)}_{*,*,w_{a,b},t_{a,b}} + \delta_{w_{a,b},t_{a,b}}}{c^{-(a,b)}_{*,*,w_{a,b},*} + \delta_{w_{a,b},*}}, \tag{10}$$

$$p(z_{a,b}|\cdot) \propto \frac{(c^{-(a,b)}_{z_{a,b},a,*,1} + \alpha_{z_{a,b}})}{(c^{-(a,b)}_{*,a,*,1} + \alpha_*)} \times \frac{(c^{-(a,b)}_{z_{a,b},*,w_{a,b},1} + \beta_{w_{a,b}})}{c^{-(a,b)}_{z_{a,b},*,*,1} + \beta_*} \tag{11}$$

where δ is defined with a scaling constant C_1 as:

$$\delta_{w_n,1} = \frac{C_1}{\log f_{w_n}}, \tag{12}$$

$$\delta_{w_n,0} = \max(0, 1 - \frac{C_1}{\log f_{w_n}}). \tag{13}$$

The two latent variables are inferred simultaneously in every Gibbs sampling iteration. The topic-labeling process is performed only when $t_{a,b} = 1$. When $t_{a,b} = 1$ for all edges, the *two-path model* becomes equivalent to the standard LDA.

4.4 Weight Model

In the *two-path model*, we assumed that there are two paths from which a writer can come. Then, we introduced the path indicator t to denote the path from which the writer comes. While a writer from the topic path is assigned with a topic, a writer from the popularity path is ignored and not assigned with a topic. We generalize this binary topic indicator t to a non-negative weight (confidence) in a *weight model*. For example, when $\tau_{obama} = 0.7$ in the *two-path model*, seven out of ten w_{obama} observations (follow edges), likely come from the topic path and the three likely come from the popularity path. Instead of probabilistically selecting which w_{obama} observation comes from which path, we may uniformly assign the τ_{obama} value to each w_{obama} observation. This τ value can be viewed as a weight or a confidence value. When we are very confident that a writer observation comes from the topic path, we may assign a value 1 to the writer observation. In the opposite case, we may assign 0 to it. If we are 70% confident, we may assign 0.7 to each writer observation.

Figure 1(e) depicts the newly introduced weight value ρ in the dotted box. ρ is associated with each writer observation and has a non-negative real number. If we strongly believe a writer is from the topic path, we may assign a high weight (even bigger than 1). Otherwise, we assign a value close to 0 or 0. Equation (14) is a ρ-incorporated version of Equation (6):

$$c_{k,m,j} = \sum_{n=1}^{N_m} I(z_{m,n} = k \& w_{m,n} = j) \cdot \rho_n, \tag{14}$$

where ρ_n is defined with a scaling constant C_2 as:

$$\rho_n = \frac{C_2}{\log f_{w_n}} \tag{15}$$

As in the *two-path model*, we believe that popular writers more likely come from the popularity path and should be assigned lower weights. When $\rho_n = 1$ for all writers, the equation becomes equivalent to that of LDA. While the *two-path model* requires the additional path-labeling process, the *weight model* is free from it and has the same complexity as LDA. The confidence and weight approach on observations can be found in the literature on recommender systems [21, 28, 32]. Especially, Steck [32] defined a new metric called Popularity-Stratified Recall by assigning lower weights to popular items, and suggested a matrix factorization method optimizing the metric. The term weighting scheme for LDA was also proposed for cross-language retrieval [39].

4.5 Other Extensions

We may further extend our models by considering reader-side popularity (in a sense that edges from a reader are more frequent in a dataset) as well as the writer-side popularity discussed so far. The reader-side popularity shows how "active" she is because it represents the length of her following list. We expect that an active reader who follows more writers can be considered less "topic-sensitive" (topic-focused) than one who follows fewer writers. If we include the reader-side popularity in the previous models, Equation (8), (10), and (15) should be accordingly extended into:

$$\begin{aligned} p(z_{a,b}|\cdot) &\propto \left(\frac{c^{-(a,b)}_{z_{a,b},*,w_{a,b}}}{c^{-(a,b)}_{z_{a,b},*,*}} + \lambda_1 \frac{c_{*,*,w_{a,b}} + \gamma_{w_{a,b}}}{c_{*,*,*} + \gamma_*}\right) \\ &\quad \times \left(\frac{c^{-(a,b)}_{z_{a,b},a,*}}{c^{-(a,b)}_{*,a,*}} + \lambda_2 \frac{c_{*,a,*} + \alpha_{z_{a,b}}}{c_{*,*,*} + \alpha_*}\right), \end{aligned} \tag{16}$$

$$\begin{aligned} p(t_{a,b}|\cdot) &\propto \frac{c^{-(a,b)}_{*,*,w_{a,b},t_{a,b}} + \delta_{w_{a,b},t_{a,b}}}{c^{-(a,b)}_{*,*,w_{a,b},*} + \delta_{w_{a,b},*}} \\ &\quad \times \frac{c^{-(a,b)}_{*,a,*,t_{a,b}} + \epsilon_{r_a,t_{a,b}}}{c^{-(a,b)}_{*,a,*,*} + \epsilon_{r_a,*}}, \end{aligned} \tag{17}$$

$$\rho_{m,n} = \frac{C_3}{\log(f_{w_n} \times f_{r_m})}, \tag{18}$$

where ϵ is a prior for reader's path indicator distribution and defined as:

$$\epsilon_{r_m,1} = \frac{C_4}{\log f_{r_m}}, \tag{19}$$

$$\epsilon_{r_m,0} = \max(0, 1 - \frac{C_4}{\log f_{r_m}}). \tag{20}$$

For the *two-path model*, we also try posing hyper priors δ' and ϵ' over δ and ϵ, respectively, similar to the approach in [37]. Then, Equation (17) should be extended to:

$$\begin{aligned} p(t_{a,b}|\cdot) &\propto \frac{c^{-(a,b)}_{*,*,w_{a,b},t_{a,b}} + \delta_* \times \frac{c_{*,*,*,t_{a,b}} + \delta'_{t_{a,b}}}{c_{*,*,*,*} + \delta'_*}}{c^{-(a,b)}_{*,*,w_{a,b},*} + \delta_*} \\ &\quad \times \frac{c^{-(a,b)}_{*,a,*,t_{a,b}} + \epsilon_* \times \frac{c_{*,*,*,t_{a,b}} + \epsilon'_{t_{a,b}}}{c_{*,*,*,*} + \epsilon'_*}}{c^{-(a,b)}_{*,a,*,*} + \epsilon_*}. \end{aligned} \tag{21}$$

We also combine the *polya-urn model* with *weight model*. There are many possible combinations in mixing two models. We report only the meaningful results in the next section.

Before moving to the next section, we summarize the symbols used in this paper in Table 1.

Table 1: Symbols used throughout this paper and their meanings

Symbol	Meaning
r	Reader (follower)
w	Writer (followed user)
z	Topic (interest)
$w_{a,b}$	Writer b in reader a's following list (follow edge)
$z_{a,b}$	Topic assigned to edge from reader a to writer b
t	Binary topic-path indicator
M	Number of unique readers
J	Number of unique writers
K	Number of unique topics
N_m	Number of writers reader m follows
f_w	In-corpus frequency of writer w
$\alpha, \beta, \gamma, \delta, \epsilon$	Dirichlet (Beta) priors
θ	Topic distribution for reader ($p(z\|r)$)
ϕ	Writer distribution for topic ($p(w\|z)$)
π	In-corpus writer distribution ($p(w)$)
τ	Topic-path distribution for writer ($p(t\|w)$)
ρ	Weight (confidence) on edge (observation)
λ	Concentration scalar

5. EXPERIMENTS

In this section, we evaluate the proposed models based on the perplexity value calculated using a real-world Twitter dataset. As baselines, we use LDA and the best performer in our prior work [7] [3]. The experimental results show that our proposed models are very effective in lowering perplexity. We also discuss why and how they perform better than the baselines.

5.1 Dataset and Experiment Settings

We use the Twitter dataset we used in our previous work [7]. It has 10 million follow edges from 14,015 reader to

[3] As perplexity is only available for probabilistic topic models, we limited our baselines to probabilistic topic models.

$2,427,373$ writers. The dataset is sampled to ensure that all the outgoing edges from a randomly sampled reader are preserved. The average number of outgoing edges for a reader is 713.52.

Table 2 summarizes the 12 representative experimental cases we report in this section. We have two baselines: *base-lda* [4], the standard LDA experiment, and *base-f2step*, the best performer in our previous work [7]. *polya-w/r/wr* denotes the *polya-urn model* experiment considering writer popularity, reader popularity (activeness), and both side popularity, respectively. *2path-w/r/wr* are cases from the *two-path model* experiments, and *wlda-w/r/wr* are from the *weight model* experiments. *p-w&w-r* denotes a combination of *polya-w* and *wlda-r*. Though we ran a lot more experiments than reported ones here, we only report some meaningful ones for clarity and to save space [5]. Other cases like the *simple model*, the *two-path model* with hyper priors, and various combination models did not show much improvements. For the *weight model*, we tested various weighting schemes based on frequency, probability, PageRank [6], pointwise mutual information (PMI) [39], etc. We only report the best weighting schemes in this section. The best weighting scheme might be different for a different corpus. In all of our experiments, we ran 100 Gibbs sampling iterations and generated 100 topic groups because they turned out reasonable in our previous experiments [7]. We also define a popular writer as a writer having more than 100 followers in our experiments [6]. We ran multiple runs to find the following optimal parameter values: $\lambda = 0.1$, $\lambda_1 = 0.2$, $\lambda_2 = 20.0$, $C_1 = 4.2$, $C_2 = 2.0$, $C_3 = 2.0$, and $C_4 = 8.6$.

Table 2: Experimental cases and descriptions

Case	Experiment Description
base-lda	LDA
base-f2step	Two-step labeling with filtering
polya-w/r/wr	Polya-urn model for writer/reader/both
2path-w/r/wr	Two-path model for writer/reader/both
wlda-w/r/wr	Weight model for writer/reader/both
p-w&w-r	*polya-w* + *wlda-r*

5.2 Perplexity Analysis

We evaluate our proposed models using the widely-used perplexity metric [5, 7, 18, 19, 40] defined as:

$$perplexity(W_{test}) = \exp^{-\frac{\sum_{w \in W_{test}} \log p(w)}{|W_{test}|}}, \qquad (22)$$

where W_{test} denotes all the writers (edges) in a test dataset. The perplexity quantifies the prediction power of a trained model by measuring how well the model handles unobserved test data. Since the exponent part of Equation (22)is a minus of the average log prediction probability over all the test edges, a lower perplexity means stronger prediction power of the model. In our previous study [7], slightly lower (3.27%) perplexity led to significant (64%) improvement on human-perceived clustering quality [7]. Thus, we believe that we can significantly improve the clustering quality of the model by further lowering its perplexity. We calculated the perplexity for two separate 10% randomly held-out datasets after training a model on the remaining 80% dataset. We averaged results from ten runs (five runs for each held-out dataset with different random seeds). As the standard LDA (*base-lda*) is designed to minimize perplexity, it is not easy to achieve lower perplexity than *base-lda*.

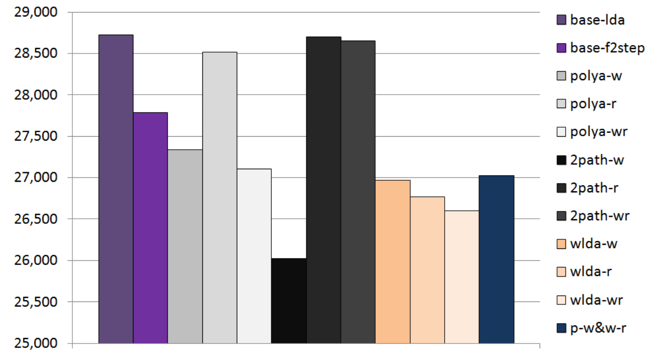


Figure 2: Perplexity comparison

We report the perplexity values from the 12 representative test cases in Figure 2, where we observe:

1. All our proposed models seem very effective in achieving lower perplexity than *base-lda*. However, the reader-side models of the *polya-urn model* and the *two-path model* show quite high perplexity compared to their counterparts (writer-side models).

2. *2path-w* shows the lowest perplexity. It shows 9.41% lower perplexity than *base-lda* (6.35% lower than *base-f2step*). However, *2path-r* and *2path-wr* show no improvements.

3. Different from the *polya-urn model* and the *two-path model*, the *weight model* shows low perplexity when the reader-side popularity is considered. It seems to conflict with our explanation that the *weight model* can be viewed as an extension to the *two-path model* in Section 4.4. We discuss this issue in Section 5.4

4. The combination models do not perform better than other models. Even *p-w&w-r*, the best performer among various combination models, shows higher perplexity than the *two-path model* and the *weight model*

If we pick the best performers in each group, the performance order would be: *two-path model* > *weight model* > combination model > *polya-urn model* > *base-f2step* > *base-lda*. In the two following sections, we investigate the *two-path model* and the *weight model* in detail.

5.3 Popularity vs. Activeness

To visualize the effect of the path-labeling, we compare the top-10 writers in two example topic groups related to *technology* from *base-lda* and *2path-w* in Figure 3. In the left

[4]We implemented our models based on the LDA implementation in [4].

[5]We tested *simple/polya/2path/wlda-w/r/wr*, *p-w/r/wr&w-w/r/wr*, and *hyperprior-w/r/wr*.

[6]We also tested 50 and 500 as the boundary value instead of 100 and the results were similar.

[7]The correlation between the perplexity and the human-perceived clustering quality was −0.806 in our previous work (excluding HLDAs).

writer	f_w	Bio
barackobama	7410	This account is run by #Obama2012 campaign staff. ...
whiteafrican	111	Where Africa and technology collide... Co-founder of Ushahidi, iHub ...
kiwanja	57	Mobile technologist. Anthropologist. Conservationist | Tech Awards ...
mrtweet	3697	We recommend awesome people to you!
stephenfry	3384	British Actor, Writer, Lord of Dance, Prince of Swimwear & Blogger
ushahidi	64	We are a non-profit tech company that specializes in developing free ...
hashtags	1862	How-to knowledge to improve your social media branding ...
stii	38	Code monkey. Love Django, Python, PHP, Surfing, Longboard ...
zoopedup	49	Zoopedup is an automotive social network that allows you ...
appfrica	41	Innovative international software development and design firm ...

(a) Example topic group from *two-path model*

writer	f_w	Bio
laughingsquid	1998	An online resource for interesting art, culture and technology. Founder ...
chrismessina	1078	Bachelor of Arts. UX designer on Google+. #godfather ...
jason	3382	LAUNCH Festival March 4-6, Hackathon March 2-4th: ...
davemorin	824	... Husband. Entrepreneur. Skier. Co-Founder and CEO of @Path
jack	1923	A sailor, a tailor *[Comment: Twitter creater Jack Dorsey]*
sarahcuda	886	I'm a reporter/author in silicon valley
dsifry	561	... Founder of Offbeat Guides, Technorati, Linuxcare, others...
anildash	1051	I love NYC, tech & funk. Cofounder of @activateinc & @thinkup. ...
joi	724	Director, MIT Media Lab
monstro	331	Co-founder Get Satisfaction, Adaptive Path, & Original. Co-author. ...

(b) When popularity-path edges are re-labeled with topics

Figure 3: Effect of path-labeling

figure, we see two highlighted writers who are not related to *technology*: *barackobama* and *stephenfry*. These celebrities are included in this group because they are so popular and followed by the people who are interested in *technology* as well. The standard LDA labels the follow edges from these people as *technology*-related even though those edges are purely generated from the popularity path. On the other hand, all the writers in the right figure are closely related to *technology*. The path-labeling reduces the chance of celebrities' appearing in irrelevant topic groups by labeling non-topic-driven follow edges with the popularity path as explained in Section 4.3.

Table 3: Path-labeling result from *2path-w*

Edge (from r to w)	*Group-p*	*Group-t*	*G-p/G-t*
Portion of edges	8.72%	91.28%	-
Avg. num. of edges to w	387.41	71.94	5.39
Avg. num. of edges from r	509.90	846.06	0.60
Avg. entropy of $p(z\|w)$	2.86	1.60	1.79
Avg. entropy of $p(z\|r)$	3.78	4.01	0.94

To measure the effect of the path-labeling, we gathered some statistics from *2path-w* and report them in Table 3. We observe that 8.72% of the test edges were processed by the popularity component (*group-p*) and 91.28% of them were processed by the topic component (*group-t*). We also calculated the average writer/reader popularity (number of incoming/outgoing edges) for the edges in both groups. The second and the third row of Table 3 tell us that each edge in *group-p* is from a reader following 509.90 writers on average [8] to a writer having 387.41 followers on average. We observe that the edges in *group-p* belong to much more popular writers than those in *group-t*. However, the edges in *group-p* seem to belong to less active readers. This finding contradicts our initial expectation on the "activeness", explained in Section 4.5, that a more active reader would be less topic sensitive. To more closely investigate this contradiction, we measured average entropy of $p(z|r)$ and $p(z|w)$. The intuition behind this measurement is that the higher

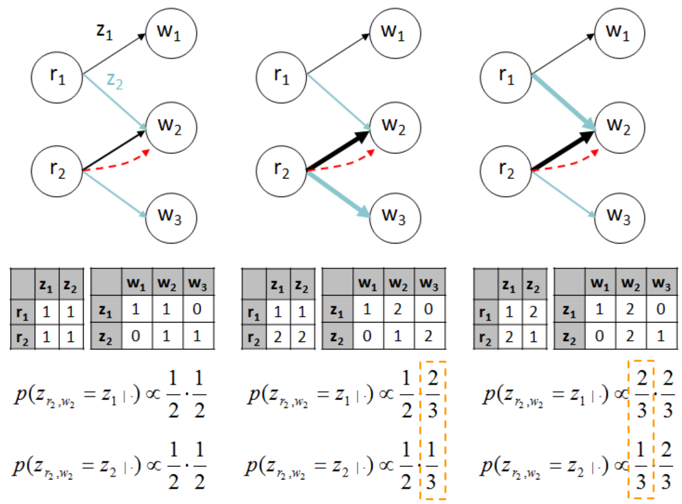


Figure 4: Reader weight vs. writer weight

entropy (uncertainty) of $p(z|r)$ or $p(z|w)$ of an edge (from reader r to writer w) may indicate that the edge has a lower probability to be labeled with a specific topic. Thus, it may have a higher chance of being labeled with the popularity path than one with a lower entropy. The fourth row of Table 3 [9] reports that the edges in *group-p* belong to the writers having much higher topic uncertainty. On the other hand, both groups show similar topic uncertainty in terms of reader-side topic entropy [10]. Thus, the reader-side popularity (activeness) does not seem to be useful in the correct path/topic-labeling. This finding also explains why *polya-r* and *2path-r* produced results with higher perplexity than their counterparts. However, we see the opposite results in the *weight model* experiments. The writer-side model (*wlda-w*) performs worse than the reader-side model (*wlda-r*). We discuss more about this anomaly in the next section.

5.4 Two-Path Model vs. Weight Model

To explain the reason why the anomaly explained in the

[8]We excluded the readers who follow less than 10 writers from our dataset because they may contain follow edges generated by pure curiosity and reciprocity. The same approach was used in [30].

[9]We used a reduced 1M-edge dataset to calculate entropy due to the limited memory size of our machine.

[10]We also observe that the reader-side entropy is much higher than the writer-side entropy.

previous section happens in the results from the *two-path model* and the *weight model*, we devise a very simple social network edge-labeling example. The upper section of Figure 4 illustrates following edges among two readers and three writers. We assume that, among the four following edges, two of them are labeled with topic z_1 (darker one), and the other two are labeled with topic z_2 (lighter one). We also assume that edges have different weights and denote an edge with a weight 1 as a narrow edge and an edge with a weight 2 as a thick edge. Thus, the bi-partite graph in the left shows the standard LDA and the one in the center shows a *wlda-r* case where edges from a reader r_2 have higher weights. The bi-partite graph in the right shows a *wlda-w* case where edges to a writer w_2 have higher weights. Each pair of tables in the middle section show a pair of count matrices (reader-topic and topic-writer) for each bi-partite graph. Each value in a count matrix is a weight assigned to each edge. Now, let's think about the case we want to label a new edge from the reader r_2 to the writer w_2 (the dotted arrow) [11]. Each bottom section shows the probability of labeling the new edge with topic z_1 or z_2 for each case according to Equation (7) [12]. Unlike the standard LDA in the left, the probabilities between topic z_1 and z_2 are different in the dotted boxes in the bottom section. Interestingly, when we assign different weights to edges from different "readers" (*wlda-r* in the center) we find that the right part of the topic-labeling probability, which corresponds to the "writer" distribution for a topic ($p(w|z)$), changes instead of the topic distribution for a reader ($p(z|r)$), and vice versa. In the topic-labeling formula given in Equation (7), the numerator in the right part is a sum of weights of the edges from many readers to a certain writer. Thus, if the weights of the edges from a reader are changed, topics associated with those edges get different association probability. However, since the numerator in the left part is a sum of weights of edges from a reader, even though those weights are changed, the sum remains the same for all topics. This aspect explains why *wlda-r* performs better than *wlda-w*. While *wlda-w* affects the topic distribution for a reader ($p(z|r)$), *wlda-r* changes the writer distribution for a topic ($p(w|z)$), which is related to writer's popularity distribution.

Though the *weight model* produces results with higher perplexity, it has the same computational complexity as the standard LDA since it does not introduce a new hidden variable. Also, we may use various weighting schemes according to the nature of application domains. Though we only reported the result from the best weighting scheme in this paper, there were many candidate weighting schemes producing results with competitive perplexity values.

6. CONCLUSION

In this paper, we proposed topic models appropriate to analyze social network graphs. Different from a textual dataset, a popular user has very important meaning in a social network dataset and should be carefully handled. We started with the *simple model* which introduces the concept of the popularity component and explored various ways to effectively incorporate it in probabilistic topic models.

In extensive experiments with a real-world Twitter dataset, our models achieved significant improvements in terms of lowering *perplexity*. Particulary, our *two-path model* showed 9.41% lower perplexity than that of LDA. Given that a 3.27% lower perplexity led to 64% higher human-perceived clustering quality in our previous work [7], we believe that our *two-path model* can also significantly improve the clustering quality. With the *two-path model*, we also showed that the reader-side popularity (activeness) is not effective in judging the reader's topic sensitivity. We extended the *two-path model* into the *weight model* and explained why the latter behaves differently from what we have expected. The *weight model* is very flexible in selecting its weighting schemes and does not increase the complexity of LDA.

Since the popularity bias is universal in various datasets including webpage visit logs, advertisement click logs, and product purchase logs, our models can effectively provide more relevant recommendations in many web services.

[11] We allow multiple edges in this example to make it simple.

[12] For simplicity, we ignored $-(a, b)$ and priors (α and β).

7. ACKNOWLEDGEMENTS

We would like to thank Christopher Moghbel, and Sunghoon Ivan Lee for their help and feedback throughout this research. We are also very grateful for valuable comments from the anonymous reviewers.

8. REFERENCES

[1] Multinomial distribution. `http://en.wikipedia.org/wiki/Multinomial_distribution`.

[2] A. Ahmed, Y. Low, M. Aly, V. Josifovski, and A. J. Smola. Scalable distributed inference of dynamic user interests for behavioral targeting. In *KDD*, pages 114–122, 2011.

[3] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. In *J. Mach. Learn. Res.*, 2008.

[4] D. Andrzejewski and X. Zhu. Latent dirichlet allocation with topic-in-set knowledge. In *Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing*, SemiSupLearn '09, pages 43–48, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[6] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 1998.

[7] Y. Cha and J. Cho. Social-network analsys using topic models. In *SIGIR*, 2012.

[8] J. Chang and D. Blei. Relational topic models for document networks. In *AIStats*, 2009.

[9] D. Cohn and H. Chang. Learning to probabilistically identify authoritative documents. In *Proceedings of the Seventeenth International Conference on Machine Learning*, ICML '00, pages 167–174, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[10] D. Cohn and T. Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. In *NIPS '00: Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, 2000.

[11] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent

semantic analysis. *Journal of the American Scociety for Information Science*, 1990.

[12] L. Dietz, S. Bickel, and T. Scheffer. Unsupervised prediction of citation influences. In *In Proceedings of the 24th International Conference on Machine Learning*, pages 233–240, 2007.

[13] E. Erosheva, S. Fienberg, and J. Lafferty. Mixed membership models of scientific publications. In *Proceedings of the National Academy of Sciences*, page 2004. press, 2004.

[14] M. Girolami and A. Kaban. On an equivalence between plsi and lda. In *SIGIR*, 2003.

[15] A. Gruber, M. Rosen-Zvi, and Y. Weiss. Latent topic models for hypertext. In *UAI 2008, Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, July 9-12, 2008, Helsinki, Finland*, pages 230–239. AUAI Press, 2008.

[16] J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *RecSys*, pages 199–206. ACM, 2010.

[17] M. Harvey, I. Ruthven, and M. J. Carman. Improving social bookmark search using personalised latent variable language models. In *WSDM*, 2011.

[18] K. Henderson and T. Eliassi-Rad. Applying latent dirichlet allocation to group discovery in large graphs. In *Proceedings of the 2009 ACM symposium on Applied Computing*, 2009.

[19] M. D. Hoffman, D. M. Blei, and F. Bach. Online learning for latent dirichlet allocation. In *In NIPS*, 2010.

[20] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.

[21] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 263–272, Washington, DC, USA, 2008. IEEE Computer Society.

[22] T. Iwata, T. Yamada, and N. Ueda. Modeling social annotation data with content relevance using a topic model. In *NIPS*, 2009.

[23] A. Java, X. Song, T. Finin, and B. L. Tseng. Why we twitter: An analysis of a microblogging community. In *WebKDD/SNA-KDD*, volume 5439 of *Lecture Notes in Computer Science*, pages 118–138. Springer, 2007.

[24] A. Mccallum, X. Wang, and A. Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of Artificial Intelligence Research*, 30:249–272, 2007.

[25] Q. Mei, D. Cai, D. Zhang, and C. Zhai. Topic modeling with network regularization. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 101–110, New York, NY, USA, 2008. ACM.

[26] 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*, 15:543–551, 2011.

[27] R. M. Nallapati, A. Ahmed, E. P. Xing, and W. W. Cohen. Joint latent topic models for text and citations. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 542–550, New York, NY, USA, 2008. ACM.

[28] R. Pan, Y. Zhou, B. Cao, N. N. Liu, R. Lukose, M. Scholz, and Q. Yang. One-class collaborative filtering. *2008 Eighth IEEE International Conference on Data Mining*, 57:502–511, 2008.

[29] N. Pathak, C. DeLong, A. Banerjee, and K. Erickson. Social topic models for community extraction. In *The 2nd SNA-KDD Workshop*, 2008.

[30] M. Pennacchiotti and S. Gurumurthy. Investigating topic models for social media user recommendation. In *Proceedings of the 20th international conference companion on World wide web*, WWW '11, pages 101–102, New York, NY, USA, 2011. ACM.

[31] J. B. Schafer, J. Konstan, and J. Riedi. Recommender systems in e-commerce. In *Proceedings of the 1st ACM conference on Electronic commerce*, EC '99, pages 158–166, New York, NY, USA, 1999. ACM.

[32] H. Steck. Item popularity and recommendation accuracy. In *Proceedings of the fifth ACM conference on Recommender systems*, RecSys '11, pages 125–132, New York, NY, USA, 2011. ACM.

[33] M. Steyvers and T. L. Griffiths. Probabilistic topic models. *Handbook of Latent Semantic Analysis*, 2007.

[34] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In *SIGKDD*, 2004.

[35] C. Sun, B. Gao, Z. Cao, and H. Li. Htm: a topic model for hypertexts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 514–522, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics.

[36] V. Tuulos and H. Tirri. Combining topic models and social networks for chat data mining. In *In Proc. of the 2004 IEEE/WIC/ACM International Conference on Web Intelligence*, 2004.

[37] H. Wallach, D. Mimno, and A. McCallum. Rethinking lda: Why priors matter. In *NIPS*, 2009.

[38] X. Wang, N. Mohanty, and A. Mccallum. Group and topic discovery from relations and text. In *In Proc. 3rd international workshop on Link discovery*, pages 28–35. ACM, 2005.

[39] A. T. Wilson and P. A. Chew. Term weighting schemes for latent dirichlet allocation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 465–473, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[40] H. Zhang, B. Qiu, C. L. Giles, H. C. Foley, and J. Yen. An lda-based community structure discovery approach for large-scale social networks. In *In IEEE International Conference on Intelligence and Security Informatics*, pages 200–207, 2007.

[41] D. Zhou, E. Manavoglu, J. Li, C. L. Giles, and H. Zha. Probabilistic models for discovering e-communities. In *World Wide Web Conference*, 2006.