Adaptive Probabilistic Topic Models for Social Networks

Arta Shayandeh

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2012

Reading Committee:

Ankur M. Teredesai, Chair

Senjuti Basu Roy

Program Authorized to Offer Degree:
Computer Science and Systems

University of Washington

Abstract

Adaptive Probabilistic Topic Models for
Social Networks

Arta Shayandeh

Chair of the Supervisory Committee:
Associate Professor Ankur M. Teredesai
Institute of Technology

Online social networks such as Twitter, LinkedIn, and Facebook generate tremendous amount of text and social interaction data. On one hand, the increasing amount of available information has motivated computational research in social network analysis to understand social structures. On the other hand, annotating, retrieving, and analyzing textual information generated within the social network is also crucial for many applications such as content ranking, recommendation systems, spam detection, and viral marketing. In this thesis we propose a composite probabilistic topic model for social networks which automatically learns topic (of interest) distributions for each entity in the social network using a combination of the available content (text) in social network and the structural properties of the network. The utility of our proposed modeling is to reduce the dimensionality of the data, exploit the underlying social structure and linkage property of the network while generating a more accurate topic model for the end-users of the social network. We discuss in detail the results on both the NIPS data set (papers from the Neural Information Processing Conference) and Enron Email (emails from large corporation) corpus. We present perplexity score for test documents as a basis of our experiments to evaluate the generalization performance of our model and provide evidence that relevant topics are discovered.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to express my gratitude to the following people, without whose encouragement and guidance this thesis would not have reached completion:

First and foremost, I acknowledge the counsel and support of my advisors, Dr. Ankur M. Teredesai. His knowledge, encouragement, and invaluable advice helped me in all the time of research and made my Master studies incredibly pleasurable. His continuous moral and academic support provided me with the motivation and ability to achieve all I have today. I am very fortunate to be supervised by him.

My appreciation extends to Dr. Senjuti Basu Roy who has always been there for me and guided me through this journey with her knowledge, ideas, and suggestions. A special thanks also goes to Dr. Donald Chinn, David Hazel and Stephen Rondeau for their unconditional help. I am grateful to have supportive labmates: Ashish Bindra, Eric Johnson and Suma Gopalakrishna whose support and friendship helped me in so many ways.

My deepest gratitude goes to my parents Manouchehr Shayandeh and Dr. Shahin Iranpour for their endless love, to my lovely brother Dr. Shahin Shayandeh for his never-ending support and incredible caring. My special thanks also goes to my love Dr. Behrouz Behmardi, who has always been there for me and this journey would be impossible without him.

Last but not least, I give my heartfelt thanks to my friends: Dr. Mina Rohani, Ava Hourmazdi and Mandana Javanmard, who always believed in me and supported me.

This thesis could not be completed without support from the Institute of Tech-

nology of University of Washington Tacoma.

DEDICATION

To my lovely parents; **Shahin** & **Manouchehr**,
for their love, endless support and encouragement.

Chapter 1

INTRODUCTION

A Social Network (SN) is defined as a network of interactions and relationships where the nodes consist of actors, and the edges consist of social relationships or interactions between these actors such as friendship, common interest, knowledge, and beliefs. In general, a network consists of a set of connected individuals which are nodes. Individuals can be a person or an organization. During the past two decades, social networks have been studied in the context of analyzing and determining the important structural patterns of interactions amongst the nodes. Many popular online social networks such as Twitter, LinkedIn, and Facebook have become increasingly popular and are extremely rich sources of content and linkage information. Moreover, this information store is growing rapidly and hence annotating, retrieving, and analyzing this information is a significant challenge. In this thesis we investigate one such challenge: building accurate topic models for users in these social networks to reflect not only their content interests but also leverages the interest of their social connections.

Social network analysis (SNA) is the study of mathematical models for mapping and measuring relationships and flows between entities such as people, organizations and groups. In SNA actors and their actions are viewed as interdependent rather than independent and relational ties(linkages) between actors are channels for transfer or flow of resources such as content(knowledge). SNA can be applied to many domains including viral marketing[16, 9], recommendation systems[7, 12], community detection[18] and information propagation[6]. Two unique characteristics of social

networks; temporal nature which requires data processing with minimum delay, and the large volume of data, make mining and managing this data even more challenging. In this thesis, we propose a probabilistic topic model for social networks which learns topic distributions of each entity in social network using messages sent between entities. Our model reduces dimensionality of the data, revealing the underlying structure, and uses the linkage property of the network. Once the topics of interest for a user are known, it helps improve the quality of results for aforementioned applications.

In this thesis we propose a probabilistic framework for modeling and analyzing social networks. The proposed framework is based on the well known Latent Dirichlet allocation (LDA) [4] model for modeling the large corpus of documents. LDA is a Bayesian probabilistic model for topic modeling. Topic model is a latent variable framework for analyzing a large corpus of text and image documents. Latent Semantic Indexing (LSI) was the first topic modeling [10] that is based on the singular value decomposition (SVD) of the term-by-document matrix. For information retrieval tasks, the representation of large amounts of text documents in a lower dimensional space found by SVD is more efficient than performing retrieval over raw unstructured data[8]. pLSI was the probabilistic extension of LSI considering a generative model for text corpora [13]. An EM algorithm was developed for inferences in pLSI. One of the main shortages of pLSI was the problem of over-fitting. This is due to the generative model in pLSI was developed in the level of documents and therefore the parameters of the model is increasing for the number of documents. Since the advent of the LDA model, significant progress has been made working on the idea of LDA and its variant for different applications. Moreover, a huge amount of work is concentrated on inferences in Bayesian network such as Gibbs sampling [19], variational Bayes [1], massage passing algorithm [15]. In this thesis we develop a variation of LDA model specific to the application of social network. Specifically, 1) we propose a novel probabilistic generative model for social networks which is suitable for capturing topics of interest for each entity in the network considering a combination of content

and network characteristics. 2) We develop a fast and accurate Bayesian inference algorithm to discover latent topics. 3) We implement the proposed algorithm and perform empirical analysis on real world datasets. We present perplexity score for test documents as a basis of our experiments to evaluate the generalization performance of our model and provide evidence that relevant topics are discovered.

Chapter 2

TECHNICAL BACKGROUND

2.1 Probability

Let's X and Y be random variables. $p(X, Y)$ is the joint distribution of both random variables. Base on the sum rule of probability, the marginal distribution of random variable X ($p(X)$), can be obtained from joint distribution using:

$$p(X) = \int_Y p(X, Y)\mathrm{d}Y \tag{2.1}$$

The conditional probability X given Y, which is commonly denoted by $P(X|Y)$, is the probability of event X if event Y have occurred. The product rule of probability states that

$$p(X, Y) = p(Y|X)p(X) \tag{2.2}$$

From the product rule and the symmetry property, $p(X, Y) = p(Y, X)$, the following relationship between the conditional probabilities is obtained which is called Bayes' theorem:

$$p(Y|X) = \frac{p(X|Y)p(Y)}{p(X)} \tag{2.3}$$

Using the sum rule, the denominator in Bayes theorem can be expressed in terms of the quantities appearing in the numerator

2.2 Generative Probabilistic Models

2.2.1 Probabilistic Models

Probabilities play a central role in modern pattern recognition. All of the probabilistic inference and learning manipulations, no matter how complex, are application of sum

and product rules [2]. Probabilistic graphical models summarize observed and latent data as random variables and the probability rules between the random variables. In a probabilistic graphical model, each node represents a random variable (or group of random variables), and the links express probabilistic relationships between these variables. The foundation of Bayesian inference follows from Bayes rule:

$$\overbrace{p(Y|X)}^{Posterior} = \frac{\overbrace{p(X|Y)}^{Likelihood}\overbrace{p(Y)}^{Prior}}{\underbrace{p(X)}_{Evidence}} \tag{2.4}$$

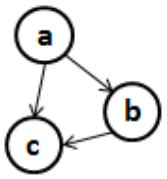
The joint distribution is interpreted as product of likelihood $p(X|Y)$ and prior $p(Y)$. The prior distribution expresses the beliefs about the random variable Y and the likelihood defines how likely the observations X are, given the prior knowledge. The conditional distribution $p(Y|X)$ is called the posterior distribution. It is the probability of the variable of interest Y, using the prior belief and observations.

2.2.2 Joint Distribution and Conditional Independence

Consider an arbitrary joint distribution $p(a, b, c)$ over three variables a, b, and c. We do not need to specify anything further about these variables, such as observed or hidden and discrete or continuous. By application of the product rule of probability, we can write the joint distribution in the form

$$p(a, b, c) = p(c|a, b)p(b|a)p(a) \tag{2.5}$$

The right hand side of Equation(2.5) can be presented in a graphical model as follows: A node for each of the random variables a, b, and c and a directed link for each conditional distribution. The result is the graph shown in Fig.2.1. Obviously the decomposition in Equation(2.5), can be chosen in a different ordering of a, b, c, and as a result we will have different graphical model. We extend the Equation(3.6) by considering the joint distribution over K variables given by $p(x_1, ..., x_k)$. So, the



$$p(a, b, c) = p(c|a, b)p(b|a)p(a)$$

Figure 2.1: A directed graphical model representing the joint probability distribution over three variables a, b, and c

joint distribution can be written as

$$p(x_1, ..., x_k) = p(x_k|x_1, ..., x_{k-1})...p(x_2|x_1)p(x_1) \tag{2.6}$$

This graph is fully connected because there is a link between every pair of nodes.

Now, let $G = (V, E)$ be a directed acyclic graph (or DAG), and let $X = (X_v)_{v \subseteq V}$ be a set of random variables indexed by V. X is a Bayesian network with respect to G if its joint probability density function (with respect to a product measure) can be written as a product of the individual density functions, conditional on their parent variables:

$$p(x) = \prod_{v \in V} p(x_v|x_{pa(v)}) \tag{2.7}$$

where $pa(v)$ is the set of parents of v (i.e. those vertices pointing directly to v via a single edge).

For any set of random variables, the probability of any member of a joint distribution can be calculated from conditional probabilities using the chain rule (given a topological ordering of X) as follows:

$$p(X_1 = x_1, ..., X_n = x_n) = \prod_{v=1}^{n} p(X_v = x_v|X_{v+1} = x_{v+1}, ..., X_n = x_n) \tag{2.8}$$

Compare this with the definition above, which can be written as:

$$p(X_1 = x_1, ..., X_n = x_n) = \prod_{v=1}^{n} p(X_v = x_v|X_j = x_j \text{ for each } X_j \text{ which is a parent } of X_v)$$

The difference between the two expressions is the conditional independence of the variables from any of their non-descendants, given the values of their parent variables.

So, X is a Bayesian network with respect to G if it satisfies the local Markov property: each variable is conditionally independent of its non-descendants given its parent variables:

$$X_v \perp\!\!\!\perp X_{V \setminus de(v)} \mid X_{pa(v)} \text{ for all } v \in V$$

where $de(v)$ is the set of descendants of v. This can also be expressed in terms similar to the first definition, as

$$p(X_v = x_v | X_i = x_i \text{ for each } X_i \text{ which is not a descendent of } of X_v) = p(X_v = x_v | X_j = x_j \text{ for each } X_j \text{ which is a paren } of X_v) \quad (2.9)$$

Note that the set of parents is a subset of the set of non-descendants because the graph is acyclic.

2.2.3 Conjugate Distributions

Bayesian models calculation is difficult, because the summations or integrals of the marginal likelihood are intractable and the strategy to facilitate model inference is to use conjugate prior distributions. Conjugate prior, $p(\phi)$, of a likelihood, $p(X|\phi)$, is a distribution that results in a posterior distribution, $p(\phi|X)$ with the same functional form as the prior but with different parameters.

2.2.4 Generative Models

When we want to draw samples from a probability distribution, Bayesian networks provide a phenomenon called generative model, which is how the observations have been generated by samples and network.

For example, assume we want to draw a sample $\hat{x_1}, ..., \hat{x_k}$ from joint distribution $p(x_1, ..., x_k)$. Suppose that the variables have been ordered such that each node has a

higher number than any of its parents. To do this, we start with the lowest-numbered node and draw a sample from the distribution $p(x1)$, which we call $\hat{x_1}$. We then work through each of the nodes in order, so that for node n we draw a sample from the conditional distribution $p(x_n|pa_n)$ in which the parent variables have been set to their sampled values. Once we have sampled from the final variable x_k, we will have achieved our objective of obtaining a sample from the joint distribution.

2.3 Inference and Parameter Estimation

A central task in the application of probabilistic models is the evaluation of the posterior distribution $p(Z|X)$ of the latent variables Z given the observed (visible) data variables X, and the evaluation of expectations computed with respect to this distribution. The model may be a fully Bayesian model in which any unknown parameters are given prior distributions and are absorbed into the set of latent variables denoted by the vector Z. For many models of practical interest, it will be infeasible to evaluate the posterior distribution or indeed to compute expectations with respect to this distribution. This could be because the dimensionality of the latent space is too high to work with directly or because the posterior distribution has a highly complex form for which expectations are not analytically tractable. For discrete variables, the marginalization involves summing over all possible configurations of the hidden variables, and though this is always possible in principle, we often find in practice that there may be exponentially many hidden states so that exact calculation is prohibitively expensive.

In such situations, we need to resort to approximation schemes, and these fall broadly into two classes,

- deterministic approximations
- stochastic approximations

Deterministic approximation is based on analytical approximations to the posterior distribution, for example by assuming that it factorizes in a particular way or

that it has a specific parametric form such as a Gaussian. As such, they can never generate exact results, and so their strengths and weaknesses are complementary of the sampling methods.

- Variational Inference: Variational Bayesian methods can provide an analytical approximation to the posterior probability of the unobserved variables, and also to derive a lower bound for the marginal likelihood of several models (i.e. the marginal probability of the data given the model, with marginalization performed over unobserved variables), with a view to performing model selection. It is an alternative to Monte Carlo sampling methods for taking a fully Bayesian approach to statistical inference over complex distributions that are difficult to directly evaluate or sample from. Variational Bayes can be seen as an extension of the EM (expectation-maximization) algorithm from maximum a-posteriori estimation (MAP estimation) of the single most probable value of each parameter, to a fully Bayesian estimation which approximately computes the entire posterior distribution of the parameters and latent variables.

- Expectation Propagation: Expectation propagation finds approximations to a probability distribution. It uses an iterative approach that leverages the factorization structure of the target distribution. It differs from other Bayesian approximation approaches such as Variational Bayesian methods.

Stochastic techniques such as Markov chain Monte Carlo, have enabled the widespread use of Bayesian methods across many domains. They generally have the property that given infinite computational resource, they can generate exact results, and the approximation arises from the use of a finite amount of processor time. In practice, sampling methods can be computationally demanding, often limiting their use to small-scale problems. Also, it can be difficult to know whether a sampling scheme is generating independent samples from the required distribution. We will described these techniques more in next section.

2.3.1 Sampling Methods

For most probabilistic models of practical interest, exact inference is intractable, and so we have to resort to some form of approximation. Here we consider approximate inference methods based on numerical sampling, also known as Monte Carlo techniques.

Although for some applications the posterior distribution over unobserved variables will be of direct interest in itself, for most situations the posterior distribution is required primarily for the purpose of evaluating expectations, for example in order to make predictions. The fundamental problem that we therefore wish to address in this section involves finding the expectation of some function $f(z)$ with respect to a probability distribution $p(z)$. The general idea behind sampling methods is to obtain a set of samples $z(l)$ (where $l = 1, ..., L$) drawn independently from the distribution $p(z)$.

For many models, the joint distribution $p(z)$ is conveniently specified in terms of a graphical model. In the case of a directed graph with no observed variables, it is straightforward to sample from the joint distribution (assuming that it is possible to sample from the conditional distributions at each node). The joint distribution is specified by

$$p(z) = \prod_{i=1}^{M} p(z_i|pa_i) \tag{2.10}$$

where z_i are the set of variables associated with node i, and pa_i denotes the set of variables associated with the parents of node i. To obtain a sample from the joint distribution, we make one pass through the set of variables in the order $z_1, ..., z_M$ sampling from the conditional distributions $p(z_i|pa_i)$. This is always possible because at each step all of the parent values will have been instantiated. After one pass through the graph, we will have obtained a sample from the joint distribution.

Now consider the case of a directed graph in which some of the nodes are instantiated with observed values. We can in principle extend the above procedure and at

each step, when a sampled value is obtained for a variable z_i whose value is observed, the sampled value is compared to the observed value, and if they agree then the sample value is retained and the algorithm proceeds to the next variable in turn. However, if the sampled value and the observed value disagree, then the whole sample so far is discarded and the algorithm starts again with the first node in the graph. This algorithm samples correctly from the posterior distribution because it corresponds simply to drawing samples from the joint distribution of hidden variables and data variables and then discarding those samples that disagree with the observed data. However, the overall probability of accepting a sample from the posterior decreases rapidly as the number of observed variables increases and as the number of states that those variables can take increases, and so this approach is rarely used in practice.

2.3.2 Markov Chain Monte Carlo

A major limitation towards more widespread implementation of Bayesian approaches is that obtaining the posterior distribution often requires the integration of high-dimensional functions. This can be computationally very difficult, but several approaches short of direct integration have been proposed (reviewed by Smith 1991, Evans and Swartz 1995, Tanner 1996). We focus here on Markov Chain Monte Carlo (MCMC) methods, which attempt to simulate direct draws from some complex distribution of interest. MCMC approaches are so-named because one uses the previous sample values to randomly generate the next sample value, generating a Markov chain.

The realization in the early 1990s (Gelfand and Smith 1990) that one particular MCMC method, the Gibbs sampler, is very widely applicable to a broad class of Bayesian problems has sparked a major increase in the application of Bayesian analysis, and this interest is likely to continue expanding for sometime to come. MCMC methods have their roots in the Metropolis algorithm (Metropolis and Ulam 1949, Metropolis et al. 1953), an attempt by physicists to compute complex integrals by expressing them as expectations for some distribution and then estimate this expec-

tation by drawing samples from that distribution.

Markov chains

Before introducing the the Gibbs sampler, a few introductory comments on Markovchains are in order. Let X_t denote the value of a random variable at time t, and let the state space refer to the range of possible X values. The random variable is a Markov process if the transition probabilities between different values in the state space depend only on the random variables current state, i.e.,

$$p(X_{t+1} = s_j | X_0 = s_k, ..., X_t = s_i) = p(X_{t+1} = s_j | X_t = s_i) \tag{2.11}$$

Thus for a Markov random variable the only information about the past needed to predict the future is the current state of the random variable, knowledge of the values of earlier states do not change the transition probability. A Markov chain refers to a sequence of random variables $(X_0, ..., X_n)$ generated by a Markov process.

2.3.3 Gibbs Sampling

In the Gibbs sampler (introduced in the context of image processing by Geman and Geman 1984), the random value is always accepted. The task remains to specify how to construct a Markov Chain whose values converge to the target distribution. The key to the Gibbs sampler is that one only considers the distribution when all of the random variables but one are assigned fixed values. Such conditional distributions are far easier to simulate than complex joint distributions and usually have simple forms. Thus, one simulates n random variables sequentially from the n univariate conditionals rather than generating a single n-dimensional vector in a single pass using the full joint distribution.

To introduce the Gibbs sampler, consider a bivariate random variable (x, y), and suppose we wish to compute one or both marginals, $p(x)$ and $p(y)$. The idea behind the sampler is that it is far easier to consider a sequence of conditional distributions,

$p(x|y)$ and $p(y|x)$, than it is to obtain the marginal by integration of the joint density $p(x, y)$, e.g., $p(x) = \int p(x, y)\mathrm{d}y$. The sampler starts with some initial value y_0 for y and obtains x_0 by generating a random variable from the conditional distribution $p(x|y = y_0)$. The sampler then uses x_0 to generate a new value of y_1, drawing from the conditional distribution based on the value x_0, $p(y|x = x_0)$. The sampler proceeds as follows

$$x_i \sim p(x|y = y_{i-1}) \tag{2.12}$$

$$y_i \sim p(y|x = x_i) \tag{2.13}$$

Repeating this process k times, generates a Gibbs sequence of length k, where a subset of points (x_j, y_j) for $1 \leq j \leq mlk$ are taken as our simulated draws from the full joint distribution.

When more than two variables are involved, the sampler is extended in the obvious fashion. In particular, the value of the kth variable is drawn from the distribution $p(\theta^{(k)}|\Theta^{-k})$ where Θ^{-k} denotes a vector containing all off the variables but k. Thus, during the ith iteration of the sample, to obtain the value of $\theta_i^{(k)}$ we draw from the distribution

$$\theta_i^{(k)} \sim p(\theta^{(k)}|\theta^{(1)} = \theta_i^{(1)}, ..., \theta^{(k-1)} = \theta_i^{(k-1)}, \theta^{(k+1)} = \theta_{i-1}^{(k+1)}, ..., \theta^{(n)} = \theta_{i-1}^{(n)}) \tag{2.14}$$

Chapter 3

ADAPTIVE TOPIC MODELS

3.1 Problem Statement

Text mining refers to the process of deriving high-quality information from text. Text mining usually involves the process of structuring the input text, deriving patterns within the structured data, and finally evaluation and interpretation of the output. On the other hand, topic model is a type of statistical model for discovering the abstract "topics" that occur in a collection of documents. Latent Dirichlet Allocation (LDA) is a probabilistic topic model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. Probabilistic topic models are "generative models" and a generative model is a model for randomly generating observable data; typically given some hidden parameters. In this section, we provide the problem definition. We first start by describing the generative process associated with the probabilistic topic model (e.g., LDA [4]) and author topic model[17] and then proceed with problem formulation.

3.2 Related Work

3.2.1 Probabilistic topic models (LDA)

Probabilistic topic models are generative models. Topic probabilities provide an explicit representation of documents in probabilistic topic model. The sampling process from this model can be explained as follows. Each document is drawn in an i.i.d. fashion. For the dth document, $d = \{1, \ldots, M\}$, a random distribution of topics $p(z_{dj} = t|\theta) \triangleq \theta_d(t)$, $t \in \{1, \ldots, T\}$ is drawn. In LDA, $\theta_d \sim \text{Dir}(\alpha)$. Then, for jth word in document d, $j = \{1, \ldots, n_d\}$, a topic assignment z_{dj} is drawn, based on the

topic distribution $\theta_d(t)$. Finally, word w_{dj} is drawn based on the conditional distribution $p(w_{dj} = l|z_{dj} = t, \Phi) \triangleq \Phi_{lt}, l = \{1, \ldots, L\}$. Note that Φ is the topic matrix where columns corresponds to topics $\{1, \ldots, T\}$ and rows corresponds to vocabulary words. (The graphical representation of LDA (known as LDA plate diagram) is shown in Fig. 3.1 and the precise sampling process for LDA is described in Algorithm 1).

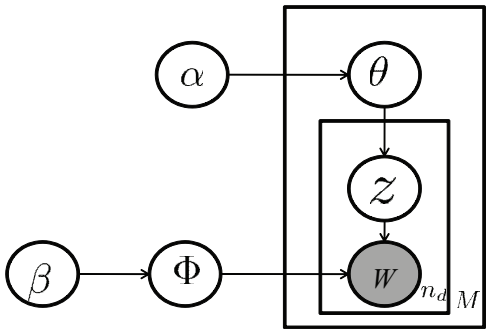


Figure 3.1: The graphical model for LDA[4].

A key observation in topic models is that the probability distribution of word w_{dj} can be obtained by marginalizing the joint word-topic distribution over the topic:

$$p(w_{dj} = l|\theta_d) = \sum_{t=1}^{T} p(w_{dj} = l|z_{dj} = t, \Phi)p(z_{dj} = t|\theta_d). \tag{3.1}$$

To simplify the notation, we represent (3.1) in a matrix format,

$$\Psi = \Phi\theta, \tag{3.2}$$

where $\Psi_{ld} \triangleq p(w_{dj} = l|\theta_d)$, $\Psi \in \mathbb{R}^{L\times M}, \Phi \in \mathbb{R}^{L\times T}$, and $\theta \in \mathbb{R}^{T\times M}$. The interpretation of LDA as a form of matrix factorization [5] is provided in Fig. 3.2 .

Latent Dirichlet Allocation is a special case of the author topic model[17], where each topic has a unique author. ϕ and θ provides information about topics in each document. However, LDA does not provide any information about the interests of authors.

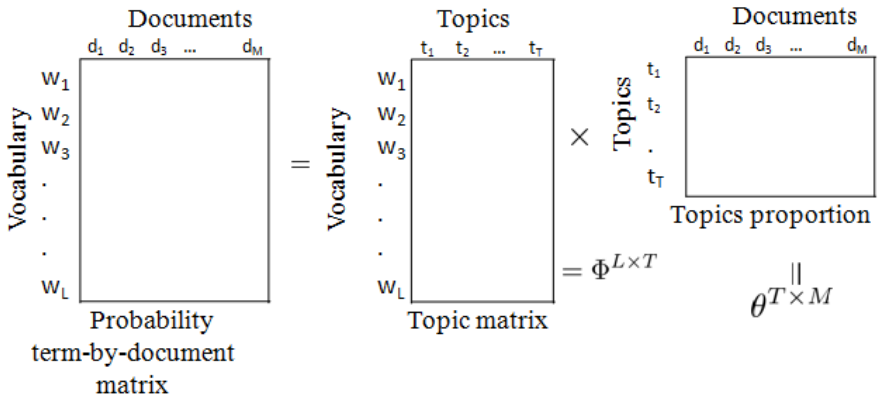


Figure 3.2: Matrix factorization interpretation of Topic Model

3.2.2 Author-Topic Model

The author topic model[17] is a hierarchical generative model in which each word w in a document is associated with two latent variables: an author, x and a topic, z. The generative process can be expressed as follow. For each author, $a = \{1, \ldots, A\}$, a random distribution of topics, θ_a,is drawn from $Dir(\alpha)$ and for each word of document d, $d = \{1, \ldots, M\}$, an author x is sampled uniformly from the set of document authors. Then a topic assignment z is drawn base on the topic distribution θ_x and finally a word is drawn base on sampled topic. The graphical representation of author topic model is shown in Fig. 3.3 and the sampling process is described in Algorithm 2.

In the author topic model, probability distribution over words for each document in a corpus is the product of three matrices (Fig. 3.4): the topic distribution over words Φ, the author distribution over topics Θ, and an $A \times D$ matrix A. The matrix A expresses the uniform distribution over authors for each document.

3.3 User topic prediction in a dynamic social network

Let us start by reviewing the social network graph shown in Fig. 3.5. Since the order of the relations in our social network are important, it is a directed graph. Let A

Algorithm 1 Generative process for LDA

for $t = 1$ to T **do**

  Draw $\Phi_t \sim \text{Dirichlet}(\beta) \in \mathbb{R}^L$

end for

for $d = 1$ to M **do**

  Draw $\theta_d \sim \text{Dirichlet}(\alpha)$

  for $j = 1$ to n_d **do**

    Draw $z_{dj} \sim \text{Discrete}(\theta_d)$

    Draw $w_{dj} \sim \text{Discrete}(\phi_{z_{dj}})$

  end for

end for

be a set of nodes in the social network and D be a set of documents (text content) authored by all users in A. In Figure 3.5, each node $a_i \in A$ represents an entity which can be a person or an organization. For each user $a \in A$ let D_a denotes a subset of documents in $D \subseteq T = \{\phi_1, \ldots, \phi_K\}$ authored by a single user $a \in A$. $\{\phi_1, \ldots, \phi_K\}$ are K topics each of which is a distribution of words over a fixed vocabulary. O_a denotes that subset of users in A whom the user a follows and similarly I_a is the set of users in A who follow a. For example, in Figure 3.5 $O_{u_1} = \{u_2, u_6, u_7, u_{10}\}$ and $I_{u_{10}} = \{u_1, u_5\}$ at time-stamp t. We refer to each user in O_a as a followee of a and in I_a as a follower of a.

Thus, for any such directed dynamic network, we are interested in modeling the topic of interest for each user. This topic modeling can be used to suggest topics of interest to entities in the social network. In the next section, we propose a solution for the problem of adaptive probabilistic topic models for social networks.

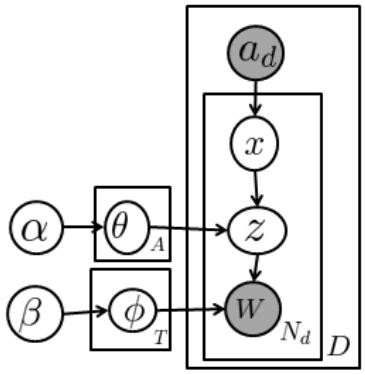


Figure 3.3: The graphical model for Author-Topic[17].

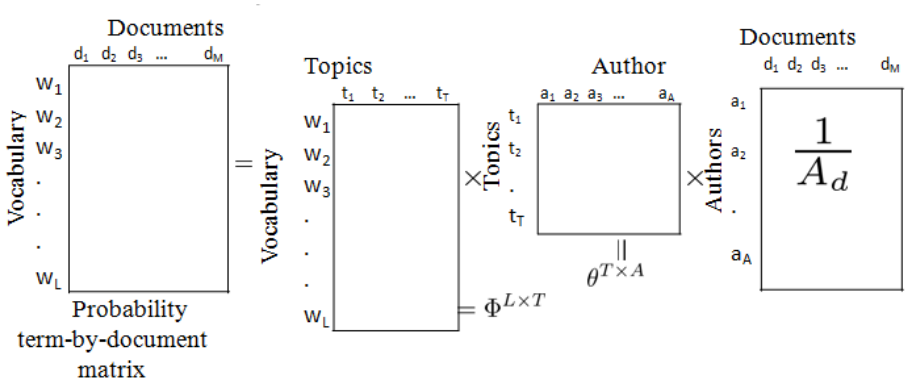


Figure 3.4: Matrix factorization interpretation of Author-Topic Model

3.4 New Model: Adaptive probabilistic topic models for Social Network

We consider an adaptive probabilistic model for the problem of user topic prediction in social network. The probabilistic model is a Bayesian hierarchical model builds on Latent Dirichlet Allocation (LDA), adding this fact that distribution over topics is not only affected by the language content of entity itself but also influenced by its network.

Algorithm 2 Generative process for Author-Topic Model

for $a = 1$ to A **do**
  Draw $\theta_a \sim$ Dirichlet(α)
end for
for $t = 1$ to T **do**
  Draw $\Phi_t \sim$ Dirichlet(β)
end for
for $d = 1$ to M **do**
  Given the authors a_d
  for $j = 1$ to n_d **do**
    Draw $x_{dj} \sim$ Uniform(a_d)
    Draw $z_{dj} \sim$ Discrete($\theta_{x_{dj}}$)
    Draw $w_{dj} \sim$ Discrete($\phi_{z_{dj}}$)
  end for
end for

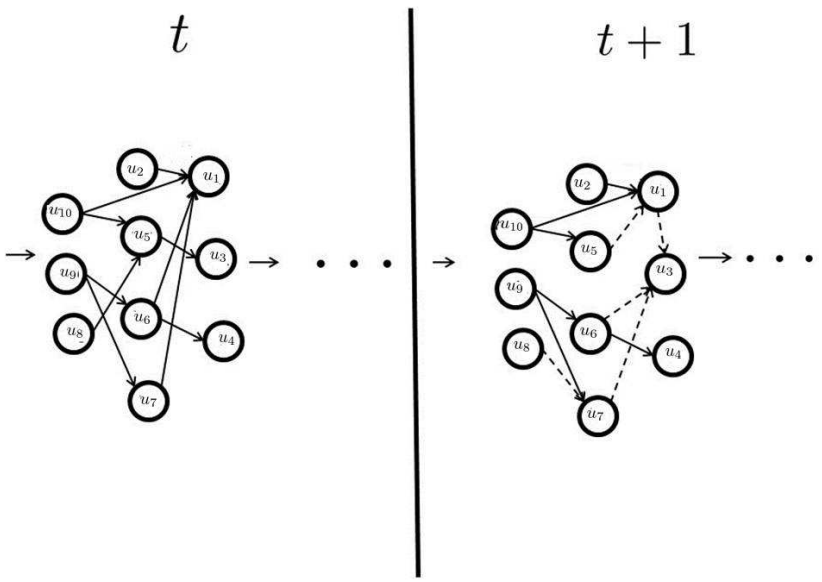
3.4.1 Generative Model

In our proposed Bayesian network generative model, first a distribution Λ is selected randomly over $O_a \cup \{a\}$ with a Drichlet prior γ and a topic distribution is selected over the list of entities with prior α. Then for each $d_{a_j} \in d_a$, an entity assignment with $p(x_{a_j}|\Lambda_a)$ and a topic assignment with $p(z_{a_j}|\theta_{a_j})$ is chosen. Then a word w is generated by randomly sampling from a topic-specific multinomial distribution ϕ_z.

In our model, each document D is assumed drawn from the following generative process:

1. Choose distribution Λ randomly over $O_a \cup \{a\}$ with a Drichlet prior γ

2. Choose user assignment $x \sim Mult(\lambda)$



(a)

Figure 3.5: Dynamic directed social network

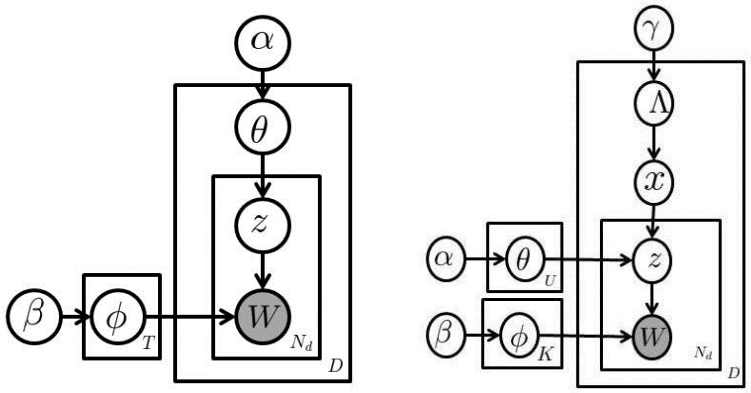
3. Choose topic proportion θ over K topics from a Drichlet(α)

4. For each word:

   (a) Choose a topic assignment $Z \sim \text{Mult}(\theta_x)$

   (b) Choose a word $W \sim Mult(\phi_z)$

The graphical model for this generative process (Algorithm 3) is shown in Fig. 3.6(b). In our model, probability distribution over words for each document is factorized to the product of three matrices (Fig. 3.7): the topic distribution over words Φ, the author distribution over topics Θ, and the document distribution over authors Λ.



(a) (b)

Figure 3.6: Graphical model representation for (a) LDA Graphical Model (b) Proposed Graphical Model. In both models, each observed word, w, is generated from a multinomial word distribution, ϕ_z, specific to a particular topic/author, z. In our proposed model, a distribution Λ is selected randomly over $O_a \cup \{a\}$ with a Drichlet prior γ.

3.4.2 Likelihoods

Base on the rule of the Bayesian network, we can specify the joint distribution of all known and hidden variables given the hyper parameters:

$$p(\Lambda, \Theta, \Phi, \vec{X}, \vec{Z}, \vec{W}|\vec{\gamma}, \vec{\alpha}, \vec{\beta}) = p(\Lambda|\vec{\gamma})p(\Theta|\vec{\alpha})p(\Phi|\vec{\beta}) \prod_{m=1}^{M} p(x_m|\Lambda) \prod_{n=1}^{N_m} p(z_{m,n}|x_m, \Theta)p(w_{m,n}|z_{m,n}, \Phi)$$

Then we can obtain the likelihood of a document, of the joint event of all words occurring, as one of its marginal distributions by integrating out the distributions Λ, Θ, Φ and summing up over X, Z:

$$\begin{aligned} p(\vec{w}_m|\vec{\gamma}, \vec{\alpha}, \vec{\beta}) &= \iiint p(\Lambda|\vec{\gamma})p(\Theta|\vec{\alpha})p(\Phi|\vec{\beta}) \\ &\quad \prod_{m=1}^{M} \sum_{x_m} p(x_m|\Lambda) \prod_{n=1}^{N_m} \sum_{z_{m,n}} p(z_{m,n}|x_m, \Theta)p(w_{m,n}|z_{m,n}, \phi)\, \mathrm{d}\Lambda \mathrm{d}\Theta \mathrm{d}\Phi \end{aligned}$$

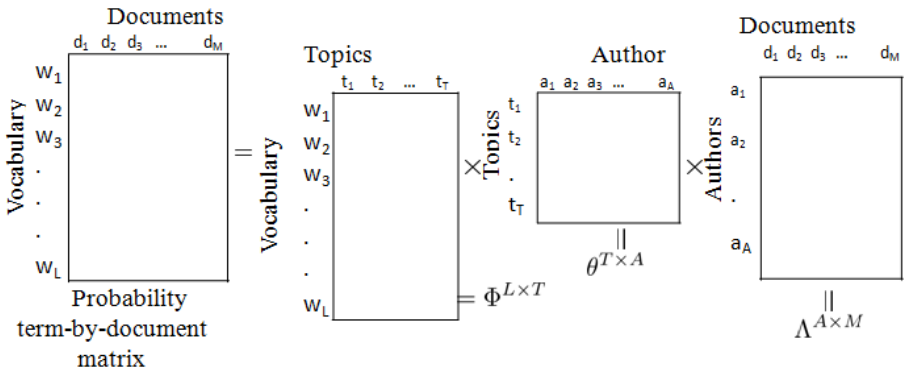


Figure 3.7: Matrix factorization interpretation of Proposed Model

Finally, the likelihood of the complete corpus is determined by the product of the likelihoods of the independent documents:

$$p(\vec{W}|\vec{\gamma},\vec{\alpha},\vec{\beta}) = \prod_{m=1}^{M} p(\vec{w}_m|\vec{\gamma},\vec{\alpha},\vec{\beta}) \tag{3.3}$$

3.5 Inference via Gibbs sampling

Exact inference for our model is intractable. In this part we introduce the potential inference algorithm which can be used to solve the probabilistic model proposed in Section 3.4. We use the Gibbs sampling as approximate inference algorithms for the solution.

To build a Gibbs sampler, the full conditionals $p(x_i|\vec{x}_{-i})$ must be found:

$$p(x_i|\vec{x}_{-i}) = \frac{p(\vec{x})}{p(\vec{x}_{-i})} = \frac{p(\vec{x})}{\int p(\vec{x})\mathrm{d}x_i} \tag{3.4}$$

For models that contain hidden variables $\vec{z}$, their posterior given the evidence, $p(\vec{z}|\vec{x})$, is a distribution commonly wanted. With Equation (3.5), the general formulation of a Gibbs sampler for such latent-variable models becomes:

$$p(z_i|\vec{z}_{-i},\vec{x}) = \frac{p(\vec{z},\vec{x})}{p(\vec{z}_{-i},\vec{x})} = \frac{p(\vec{z},\vec{x})}{\int p(\vec{z},\vec{x})\mathrm{d}z_i} \tag{3.5}$$

Algorithm 3 Generative Model

{Topic Plate}
for all topics $k \in [1, K]$ **do**
  Sample $\phi_k \sim Dir(\vec{\beta})$
end for
{Author Plate}
for all authors $a \in [1, A]$ **do**
  Sample $\theta_a \sim Dir(\vec{\alpha})$
end for
{Document Plate}
for all documents $m \in [1, M]$ **do**
  Sample $\lambda_m \sim Dir(\vec{\gamma})$
  Sample author $x_m \sim Mult(\lambda_m)$
  for all words $n \in [1, N_m]$ in document m **do**
    Sample topic $z_{m,n} \sim Mult(\theta_{x_m})$
    Sample word $w_{m,n} \sim Mult(\phi_{z_{m,n}})$
  end for
end for

where the integral changes to a sum for discrete variables.

To derive a Gibbs sampler for our model, we apply the hidden-variable method described above. The hidden variables in our model are x_m and $z_{m,n}$. We do not need to include the parameter sets Λ, Θ and Φ because they can be interpreted as statistics of the associations between the observed $w_{m,n}$ and the corresponding $x_m, z_{m,n}$, the state variables of the Markov chain. The strategy of integrating out some of the parameters for model inference is often referred to as collapsed Gibbs sampling [14].

3.5.1 Joint distribution

In our model, this joint distribution can be factored:

$$p(\vec{W}, \vec{Z}, \vec{X}|\vec{\alpha}, \vec{\beta}, \vec{\gamma}) = p(\vec{W}|\vec{Z}, \vec{\beta})p(\vec{Z}|\vec{X}, \vec{\alpha})p(\vec{X}|\vec{\gamma}) \tag{3.6}$$

Because the first term independent from $\vec{\alpha}, \vec{\gamma}$ given Z, second term is independent from $\vec{\beta}, \vec{\gamma}$ given X and third term is independent of $\vec{\alpha}, \vec{\beta}$ we can handle each term separately.

The first term, $p(\vec{W}|\vec{Z}, \vec{\beta})$ can be derived from

$$p(\vec{W}|\vec{Z}, \vec{\beta}) = \int p(\vec{W}|\vec{Z}, \Phi)p(\Phi|\beta)\mathrm{d}\Phi \tag{3.7}$$

and $p(\vec{W}|\vec{Z}, \Phi)$ can be derived from a multinomial on the observed word counts given the associated topic:

$$\begin{aligned} p(\vec{W}|\vec{Z}, \Phi) &= \prod_{i=1}^{W} p(w_i|z_i) \\ &= \prod_{i=1}^{W} \phi_{z_i, w_i} \end{aligned} \tag{3.8}$$

That is the W words of corpus are observed according to independent multinimial trails with parameters conditioned on the topic z_i.

Now we can split the product over words into product over topics and vocabulary.

$$\begin{aligned} p(\vec{W}|\vec{Z}, \Phi) &= \prod_{k=1}^{K} \prod_{i:z_i=k} p(w_i = t|z_i = k) \\ &= \prod_{k=1}^{K} \prod_{t=1}^{V} \phi_{k,t}^{n_k^{(t)}} \end{aligned} \tag{3.9}$$

where $n_k^{(t)}$ is number of times term t is observed with topic k. Given Equation(3.9) and the fact $\Phi \sim Dirichlet(\beta)$

$$p(\vec{W}|\vec{Z}, \vec{\beta}) = \int p(\vec{W}|\vec{Z}, \Phi)p(\Phi|\beta)\mathrm{d}\Phi$$

$$
\begin{aligned}
&= \int \prod_{z=1}^{K} \prod_{t=1}^{V} \phi_{z,t}^{n_z^{(t)}} \prod_{z=1}^{K} \frac{1}{B(\vec{\beta})} \prod_{t=1}^{V} \phi_{z,t}^{\beta_{(t)}-1} \mathrm{d}\Phi \\
&= \int \prod_{z=1}^{K} \frac{1}{B(\vec{\beta})} \prod_{t=1}^{V} \phi_{z,t}^{n_z^{(t)}+\beta_{(t)}-1} \mathrm{d}\Phi \\
&= \prod_{z=1}^{K} \frac{B(\vec{n}_z + \beta)}{B(\vec{\beta})} \qquad (3.10)
\end{aligned}
$$

where $\vec{n}_z = \{n_z^{(t)}\}_{t=1}^{V}$. The topic distribution can be derived

$$
\begin{aligned}
p(\vec{Z}|\vec{X}, \Theta) &= \prod_{i=1}^{W} p(z_i|x_{d_i}) \\
&= \prod_{i=1}^{W} \theta_{z_i, x_{d_i}} \qquad (3.11)
\end{aligned}
$$

That is the topic of W words of corpus are according to independent multinomial trail with parameters condition on author x_{d_i} (author of document d_i).

$$
\begin{aligned}
p(\vec{Z}|\vec{X}, \Theta) &= \prod_{a=1}^{A} \prod_{k=1}^{K} p(z_i = k|x_{d_i} = a) \\
&= \prod_{a=1}^{A} \prod_{k=1}^{K} \theta_{a,k}^{n_a^{(k)}} \qquad (3.12)
\end{aligned}
$$

where $n_a^{(k)}$ is number of times topic k is assigned to author a. Given Equation(3.12) and the fact $\Theta \sim Dirichlet(\alpha)$

$$
\begin{aligned}
p(\vec{Z}|\vec{X}, \vec{\alpha}) &= \int p(\vec{Z}|\vec{W}, \Theta) p(\Theta|\alpha) \mathrm{d}\Theta \\
&= \int \prod_{a=1}^{A} \frac{1}{B(\vec{\alpha})} \prod_{k=1}^{K} \theta_{a,k}^{n_a^{(k)}+\alpha_{(k)}-1} \mathrm{d}\Theta \\
&= \prod_{a=1}^{A} \frac{B(\vec{n}_a + \alpha)}{B(\vec{\alpha})} \qquad (3.13)
\end{aligned}
$$

where $\vec{n}_a = \{n_a^{(k)}\}_{k=1}^{K}$. The author distribution can be derived the same way.

$$
p(\vec{X}|\Lambda) = \prod_{m=1}^{M} \prod_{a=1}^{A} p(x_{d_i} = a|d_i = m)
$$

$$= \prod_{m=1}^{D}\prod_{a=1}^{A}\lambda_{m,a}^{n_m^{(a)}} \tag{3.14}$$

where $n_m^{(a)}$ is number of times user a is assigned to document m. Given Equation(3.14) and the fact $\Lambda \sim Dirichlet(\gamma)$

$$\begin{aligned} p(\vec{X}|\vec{\gamma}) &= \int p(\vec{X}|\Lambda)p(\Lambda|\gamma)\mathrm{d}\Lambda \\ &= \int \prod_{m=1}^{D}\frac{1}{B(\vec{\gamma})}\prod_{a=1}^{A}\lambda_{m,a}^{n_m^{(a)}+\gamma_{(a)}-1}\mathrm{d}\Lambda \\ &= \prod_{m=1}^{D}\frac{B(\vec{n}_m+\gamma)}{B(\vec{\gamma})} \end{aligned} \tag{3.15}$$

where $\vec{n}_m = \{n_m^{(a)}\}_{a=1}^{A}$. So, the joint distribution becomes:

$$p(\vec{W},\vec{Z},\vec{X}|\vec{\alpha},\vec{\beta},\vec{\gamma}) = \prod_{z=1}^{K}\frac{B(\vec{n}_z+\beta)}{B(\vec{\beta})}\prod_{a=1}^{A}\frac{B(\vec{n}_a+\alpha)}{B(\vec{\alpha})}\prod_{m=1}^{D}\frac{B(\vec{n}_m+\gamma)}{B(\vec{\gamma})} \tag{3.16}$$

3.5.2 Full Conditional

From the joint distribution, we can derive the update equation from which the Gibbs sampler draws the hidden variable. Using the chain rule and the fact that $\vec{X} = \{x_{d_i} = a, \vec{x}_{-d_i}\}$ and $\vec{Z} = \{\{z_i\}_{i:w_i \text{ in } {}_{d_i}}, z_{-i}\}$ yields:

$$\begin{aligned} p(x_{d_i} = a|\vec{x}_{-d_i},\vec{Z},\vec{W}) &= \frac{p(\vec{X},\vec{Z},\vec{W})}{p(\vec{x}_{-d_i},\vec{Z},\vec{W})} \\ &= \frac{p(\vec{W}|\vec{Z})p(\vec{Z}|\vec{X})p(\vec{X})}{p(\vec{W}|\vec{Z})p(\vec{Z}|\vec{x}_{-d_i})p(\vec{x}_{-d_i})} \\ &= \frac{p(\vec{Z}|\vec{X})}{p(\vec{z}_{-d_i}|\vec{x}_{-d_i})p(\vec{z}_{d_i})}\frac{p(\vec{X})}{p(\vec{x}_{-d_i})} \\ &\propto \frac{p(\vec{Z}|\vec{X})}{p(\vec{z}_{-d_i}|\vec{x}_{-d_i})}\frac{p(\vec{X})}{p(\vec{x}_{-d_i})} \\ &\propto p(\vec{z}_{d_i}|x_{d_i})\frac{p(\vec{X})}{p(\vec{x}_{-d_i})} \\ &\propto \frac{p(\vec{X})}{p(\vec{x}_{-d_i})} \end{aligned}$$

$$
\begin{aligned}
&\propto \frac{B(\vec{n}_m + \vec{\gamma})}{B(\vec{n}_{m,-d_i} + \vec{\gamma})} \\
&\propto \frac{\Gamma(n_m^{(a)} + \gamma_a)\Gamma(\sum_{a=1}^{A} n_{m,-d_i}^{a} + \gamma_a)}{\Gamma(n_{m,-d_i}^{(a)} + \gamma_a)\Gamma(\sum_{a=1}^{A} n_m^{a} + \gamma_a)} \\
&\propto \frac{n_{m,-d_i}^{(a)} + \gamma_a}{\sum_{a=1}^{A} n_{m,-d_i}^{(a)} + \gamma_a} \qquad (3.17)
\end{aligned}
$$

where $\vec{z}_{d_i}$ is the topic of words in document d_i and the counts $n_{.,-d_i}^{(.)}$ indicate that the document d_i is excluded.

$$
\begin{aligned}
p(z_i = k|\vec{z}_{-i}, \vec{W}, \vec{X}) &= \frac{p(\vec{X}, \vec{Z}, \vec{W})}{p(\vec{X}, \vec{z}_{-i}, \vec{W})} \\
&= \frac{p(\vec{W}|\vec{Z})}{p(\vec{w}_{-i}|\vec{z}_{-i})p(w_i)} \frac{p(\vec{Z}|\vec{X})}{p(\vec{z}_{-i}|\vec{X})} \\
&\propto \frac{p(\vec{W}|\vec{Z})}{p(\vec{w}_{-i}|\vec{z}_{-i})} \frac{p(\vec{Z}|\vec{X})}{p(\vec{z}_{-i}|\vec{X})} \\
&\propto \frac{B(\vec{n}_z + \vec{\beta})}{B(\vec{n}_{z,-i} + \vec{\beta})} \frac{B(\vec{n}_a + \vec{\alpha})}{B(\vec{n}_{a,-i} + \vec{\alpha})} \\
&\propto \frac{\Gamma(n_z^{(t)} + \beta_t)\Gamma(\sum_{t=1}^{V} n_{z,-i}^{t} + \beta_t)}{\Gamma(n_{z,-i}^{(t)} + \beta_t)\Gamma(\sum_{t=1}^{V} n_z^{t} + \beta_t)} \frac{\Gamma(n_a^{(k)} + \alpha_k)\Gamma(\sum_{k=1}^{K} n_{a,-i}^{k} + \alpha_k)}{\Gamma(n_{a,-i}^{(k)} + \alpha_k)\Gamma(\sum_{k=1}^{K} n_a^{k} + \alpha_k)} \\
&\propto \frac{n_{z,-i}^{(t)} + \beta_t}{\sum_{t=1}^{V} n_{z,-i}^{(t)} + \beta_t} \frac{n_{a,-i}^{(k)} + \alpha_k}{\sum_{k=1}^{K} n_{a,-i}^{(k)} + \alpha_k} \qquad (3.18)
\end{aligned}
$$

where the counts $n_{.,-i}^{(.)}$ indicate that the word i is excluded from document and topic.

3.5.3 Multinomial parameters

Equation(3.18) is the conditional probability derived by marginalizing out the random variables ϕ (the probability of word given a topic) and θ (the probability of topic given author) and Equation(3.17) is the conditional probability derived by marginalizing out the random variables λ (the probability of author give document). These random

variables are estimated from samples via

$$\phi_{kt} = \frac{n_k^{(t)} + \beta_t}{\sum_{t=1}^{V} n_k^{(t)} + W\beta} \tag{3.19}$$

$$\theta_{ak} = \frac{n_a^{(k)} + \alpha_k}{\sum_{k=1}^{K} n_a^k + K\alpha} \tag{3.20}$$

$$\lambda_{ma} = \frac{n_m^{(a)} + \gamma_a}{\sum_{a=1} A n_m^a + A\gamma} \tag{3.21}$$

Where $n_k^{(t)}$ is the number of times word t is assigned to topic k, $n_a^{(k)}$ is the number of times topic k is assigned to author a and n_m^a is the number of times author a is assigned to document m.

3.5.4 Gibbs Sampling Algorithm

Using Eqs. (3.17), (3.18), (3.19), (3.20) and (3.21), the Gibbs sampling procedure in Algorithm 4 can be run. The procedure itself uses only 8 larger data structures, the count $n_k^{(t)}$, $n_a^{(k)}$, n_m^a variables, which have dimension $K \times V$, $A \times K$ and $M \times A$ respectively, their row sums n_z, n_a and n_m with dimension K, A and M, as well as the state variable $z_{m,n}$ with dimension W and x_m with dimension M.

Table 3.1: Quantities in the Model

Quantities	explanation
M	number of documents
K	number of topics
V	number of terms in vocabulary
A	number of authors
$\vec{\alpha}$	hyperparameter
$\vec{\beta}$	hyperparameter
$\vec{\gamma}$	hyperparameter
$\vec{\lambda}_m$	parameter notation for $p(x\|d = m)$, the author mixture proportion of document m
$\vec{\theta}_a$	parameter notation for $p(z\|x = a)$, the topic mixture proportion for author a
$\vec{\phi}_k$	parameter notation for $p(t\|z = k)$, the mixture component of topic k
N_m	length of document m
x_m	author indicator for document m
$z_{m,n}$	topic indicator for nth word in document m
$w_{m,n}$	term indicator for nth word in document m

Algorithm 4 Gibbs Sampling Algorithm

{Initialization}

for all documents $m \in [1, M]$ **do**

Sample author $x_m = a \sim Mult(1/A)$

increment documentauthor count: n_m^a and documentauthor sum: n_m

for all words $n \in [1, N_m]$ in document m **do**

Sample topic $z_{m,n} = k \sim Mult(1/K)$

increment author-topic count: n_a^k and author-topic sum: n_a

increment topic-word count: n_k^n and topic-word sum: n_z

end for

end for

{Gibbs sampling over burn-in period and sampling period}

while not finished **do**

for all documents $m \in [1, M]$ **do**

decrement counts and sums document-author

Sample author x_m according to (3.17)

increment documentauthor count: n_m^a and documentauthor sum: n_m

for all words $n \in [1, N_m]$ in document m **do**

decrement counts and sums topic-word, author-topic

Sample topic $z_{m,n}$ according to (3.18)

increment author-topic count: n_a^k and author-topic sum: n_a

increment topic-word count: n_k^n and topic-word sum: n_z

end for

end for

if converged and L sampling iterations since last read out **then**

read out parameter set ϕ , θ and λ according to (3.19),(3.20),(3.21)

end if

end while

Chapter 4

EXPERIMENTS

In this part, we illustrate the applicability of the proposed framework for topic modeling in social network in two real datasets.

4.1 Datasets

For our analysis, we consider NIPS and Enron email datasets. NIPS datasets is a set of papers from 13 years (1987 to 1999) of the Neural Information Processing (NIPS) Conference[1]. This data set contains $M = 1,740$ papers, $A = 2,037$ different authors, a total of $N = 2,301,375$ word tokens, and a vocabulary size of $V = 13,649$ unique words. The second corpus is the Enron email data set[2], where it contains a set of $D = 121,298$ emails, with $A = 11,195$ unique authors, and $N = 4,699,573$ word tokens. For the purpose of our analysis, we consider a sub-sample of each datasests as follows:

NIPS data set: For our experiments we randomly select $D = 500$ papers from the data set which contains $A = 886$ different authors, total of $N = 658,876$ word tokens and vocabulary size $V = 13,649$. We preprocessed each documents by removing stop words from a standard list.

Enron Email Data set: For the purpose of our experiments we create a data set which contains the social network from the original data set in this manner. We use email directories of seven users. The users are: Sally Beck (Chief Operating Officer), Darren Farmer (Logistics Manager), Vincent Kaminski (Head of Quantitative

[1]Available on-line at http://www.cs.toronto.edu/ roweis/data.html

[2]Available on-line at http://www-2.cs.cmu.edu/ enron/

Modeling Group), Louise Kitchen (President of EnronOnline), Michelle Lokay (Administrative Assistant), Richard Sanders (Assistant General Counsel) and William Williams III (Senior Analyst). For each of these targeted users, we select the emails they send and receive (For the received emails we consider those emails with the sender who sends more than total of 20 emails to the user). For each of received emails we put the sender of email as the author of the email. For each targeted user i we have a set of users who send emails to user i (O_i). For those emails that are sent by targeted user i, we put user i and the set of users in O_i as author. By this way, we have our directed social graph. We preprocessed each documents by removing stop words from a standard list and all words with global occurrence less than 10. The new data set contains $D = 11,983$ emails, $A = 138$ different authors, a total $N = 1,195,893$ word tokens and vocabulary size $V = 4,662$.

We consider two different set of experiments in our analysis. First, we evaluate the generalization performance of our model using perplexity. Moreover, we show the convergence analysis of the proposed inference algorithm using perplexity. The perplexity score is calculated as follws:

$$perplexity(w_d|a_d, D_{train}) = \exp\left(-\frac{\log p(w_d|a_d, D_{train})}{N_d}\right) \tag{4.1}$$

Where $p(w_d|a_d, D_{train})$ is the probability assigned by the model to the w_d in test document and N_d is the number of word in test document. For multiple test documents:

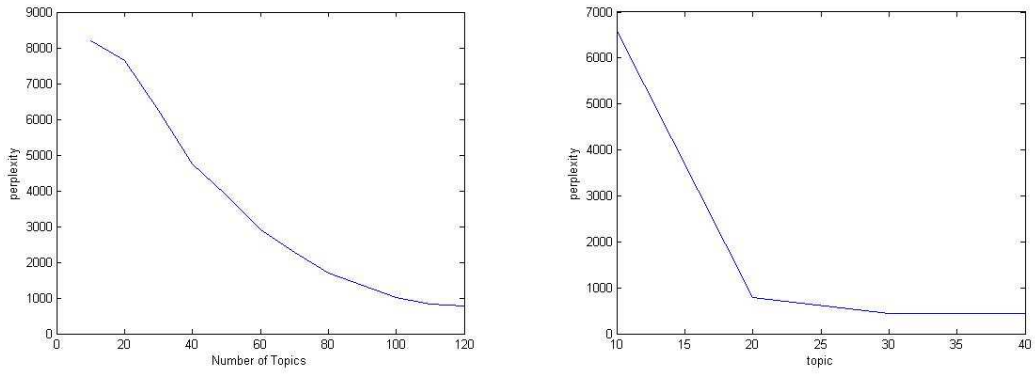
$$perplexity(D_{test}) = \frac{\sum_{d=1}^{D_{test}} perplexity(w_d|a_d, D_{train})}{D_{test}} \tag{4.2}$$

Perplexity is typically monotonically decreasing with respect to the likelihood of the test data and hence a lower perplexity score indicates better performance.

In the second part, we perform the similarity analysis on the distribution of topics for each user and show how the model can be used to propose topics to each user.

4.2 Generalization performance using perplexity

Fig. 4.1 shows the perplexity plotted against the number of hidden topics K for both NIPS and Enron data set. It is obvious that the perplexity decreases with an increasing number of topics. If the number of topics is small, i.e. $K \leq 60$ for NIPS data set and $K \leq 20$ for Enron data set, the perplexity grows rapidly indicating that the model does not fit the unseen training data. Then the perplexity stabilized. The lower perplexity means the better fitted model. The general observation indicates that perplexity as a function of number of topics gets smaller as we consider more topics in our model. It suggests that as the model gets richer in terms of the parameters, its generalization performance increases. However, after some point increasing the number of topics does not add more value to the model and in fact its generalization performance remains constant (perplexity is constant).



(a) (b)

Figure 4.1: Perplexity against Number of Topic (a) NIPS Data Set (b) Enron Data Set

4.3 Convergence analysis of Gibbs sampling

The convergence of the Markov chain used to sample a set of variables is a common issue that arises in applying MCMC techniques. The issue have two different sides:

1. Figure out when the performance of a model trained by sampling begins to stabilize.

2. when the Markov chain actually reaches the posterior distribution.

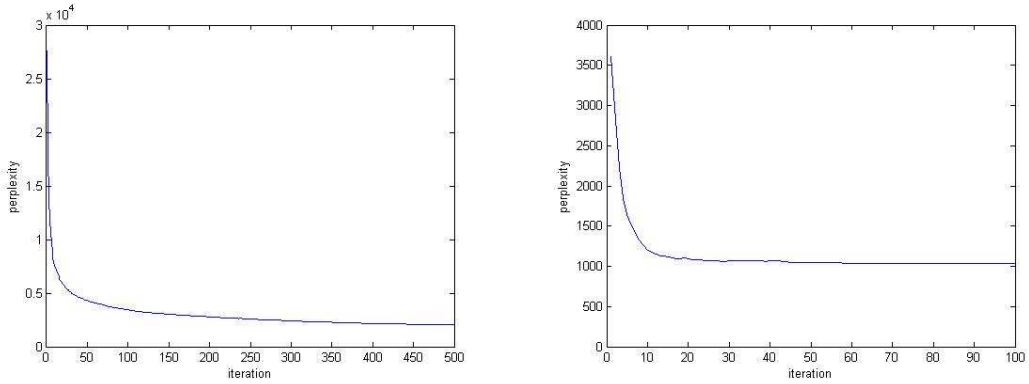
In general, for real data sets, there is no proof method for answering the latter question[17]. So, we just focus on the former, using the perplexity of the model on test documents to evaluate when the performance of the model begins to stabilize.

Fig. 4.2(a) shows perplexity as a function of the number of iterations of the Gibbs sampler, for a model with 60 topics fit to the NIPS data. Samples after i iterations (where i is the x-axis in the graph) are used to produce a perplexity score on test documents ($D_{test} = \frac{1}{3}M$). In the experiments we do not estimate the hyper-parameters α, β and γ. They are fixed at 1, 0.01 and 1 in each of the experiments. It appears from the Fig. 4.2(a) that performance of models trained using the Gibbs sampler appears to stabilize quickly (after about 50 iterations). Qualitatively similar results were obtained for the Enron email data set (Fig. 4.2(b)), the perplexity values stabilized after a 10 iterations of the Gibbs sampler.

4.4 Topic Discovery

In this section, we showed the topic discovered by our model on the Enron Email data set and compare it with topic discovered by author topic model.

First we run author topic model on the emails of 7 targeted users, we explained on Dataset section. We provide the results for one of our targeted users, Sally.beck. Author topic model assigned topics showed in Fig.4.3 to Sally.beck. We plot author



(a) (b)

Figure 4.2: Perplexity against Number of Iteration (a) NIPS Data Set (b) Enron Data Set

topic distribution of Sally.beck and all users in her network (Fig.4.4). It is obvious in Fig.4.4 most users in her network talked about topics showed in Fig.4.5.

Topic 7-operation		Topic 9-Grant Proposal	
corporation	0.0717	sale	0.0193
subject	0.0404	original	0.0118
operations	0.0248	agreement	0.0094
development	0.0195	plans	0.0086
enron	0.0125	detail	0.0084

Figure 4.3: Topics Assigned to Sally.beck with Author Topic Model

Running our model on the same data set, topic in Fig.4.6 is assigned to Sally.beck. It is obvious topics in Fig.4.6 are the combination of topics both in Fig.4.3 and Fig.4.5. So, both Sally.beck topic of interest and also topic of interest of users in her network are assigned to her using our model. We get the same results for other targeted users.

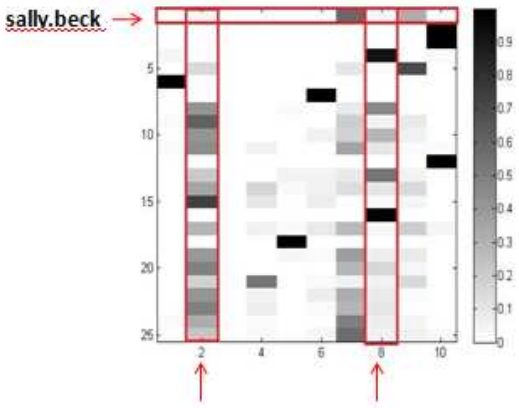


Figure 4.4: Author Topic distribution to Sally.beck and all users in her network using Author Topic model

Topic 2-Meeting Setup		Topic 8-Team Report	
information	0.0242	business	0.0425
meeting	0.0212	employees	0.0410
group	0.0163	time	0.0186
global	0.0139	team	0.0176
trading	0.0115	report	0.0150

Figure 4.5: Topics Assigned to most of users in network of Sally.beck with Author Topic Model

Topic 2		Topic 4		Topic 7	
deal	0.0247	time	0.0233	meet	0.0193
origin	0.0113	work	0.0174	corporation	0.0118
subject	0.0097	manage	0.0171	sale	0.0094
corporation	0.0093	team	0.0092	group	0.0086
sale	0.0086	report	0.0083	business	0.0084

Figure 4.6: Topics Assigned to Sally.beck with our proposed adaptive topic model

Chapter 5

FUTURE RESEARCH & CONCLUSIONS

The model proposed in this thesis provides a relatively simple probabilistic model for exploring the topics in social network. The primary benefit of this model is that it allows us to include the network information available in social network in topic model to get more effective results for authors topic distribution. We provide results of applying our model to NIPS and Enron emails datasets to evaluate the generalization performance and show convergence analysis using perplexity.

As we show in Fig. 3.5, the social network graph is dynamic and the set of followers and followees will change in the social network over time. For example, in Fig. 3.5, $O^t_{u_1} = \{u_2, u_6, u_7, u_{10}\}$ at time-stamp t and $O^{t+1}_{u_1} = \{u_2, u_5, u_{10}\}$ at time-stamp $t+1$. Time is measured based on a time scale such as hours, days, and months specific to the nature of the network under study. Dotted lines in the figure show the change of followers and followees from time-stamp t to $t+1$.

Topics of a given entity can get affected by (i) the dynamic nature of relations in the network and by (ii) the evolution of topics over the time. The proposed model should capture the variation in network structure as well as topic evolution that can be the topic of future researches. To consider the changes of the network during time, one idea is to have the Bayesian network like dynamic topic model[3]. So, the topic distribution will evolve during time(Fig. 5). Another idea is to use a Markov model[11].

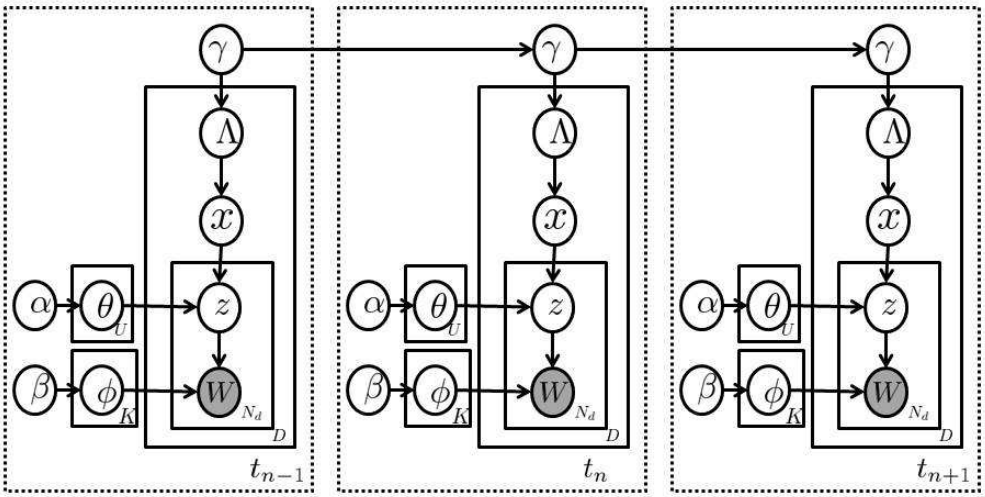


Figure 5.1: Graphical model representation of our proposed dynamic model. Topic distribution will evolve during time by changing the network.

BIBLIOGRAPHY

[1] Matthew J. Beal. *Variational algorithms for approximate Bayesian inference*. 2003.

[2] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 1st ed. 2006. corr. 2nd printing edition, October 2006.

[3] David M. Blei and John D. Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pages 113–120, New York, NY, USA, 2006. ACM.

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

[5] John Canny. Gap: a factor model for discrete data. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 122–129, New York, NY, USA, 2004. ACM.

[6] Meeyoung Cha, Alan Mislove, and Krishna P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. In *Proceedings of the 18th international conference on World wide web*, WWW '09, pages 721–730, New York, NY, USA, 2009. ACM.

[7] Jilin Chen, Rowan Nairn, Les Nelson, Michael Bernstein, and H. Chi. Short and tweet: Experiments on recommending content from information streams.

[8] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[9] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '01, pages 57–66, New York, NY, USA, 2001. ACM.

[10] S. T. Dumais, G. W. Furnas, T. K. Landauer, S. Deerwester, and R. Harshman. Using latent semantic analysis to improve access to textual information. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '88, pages 281–285, New York, NY, USA, 1988. ACM.

[11] Amit Gruber, Michal Rosen-zvi, and Yair Weiss. Hidden topic markov models. In *In Proceedings of Artificial Intelligence and Statistics*, 2007.

[12] John Hannon, Mike Bennett, and Barry Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *Proceedings of the fourth ACM conference on Recommender systems*, RecSys '10, pages 199–206, New York, NY, USA, 2010. ACM.

[13] T. Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

[14] J. Liu. The collapsed Gibbs sampler in Bayesian computations with application to a gene regulation problem. *Journal of the American Statistical Association*, 89(958–966), 1994.

[15] Thomas P. Minka. Expectation propagation for approximate bayesian inference. In *UAI '01: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence*, pages 362–369, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[16] Matthew Richardson and Pedro Domingos. Mining knowledge-sharing sites for viral marketing. pages 61–70. ACM Press, 2002.

[17] Michal Rosen-Zvi, Thomas Griffiths, Mark Steyvers, and Padhraic Smyth. The author-topic model for authors and documents.

[18] Joshua R. Tyler, Dennis M. Wilkinson, and Bernardo A. Huberman. *Email as spectroscopy: automated discovery of community structure within organizations*, pages 81–96. Kluwer, B.V., Deventer, The Netherlands, The Netherlands, 2003.

[19] B. Walsh. Markov chain monte carlo and gibbs sampling, 2004.