


12-2015

# Probabilistic Graphical Modeling on Big Data

Ming-Hua Chung

*University of Arkansas, Fayetteville*

Follow this and additional works at: <http://scholarworks.uark.edu/etd>

 Part of the [Applied Statistics Commons](#), and the [Categorical Data Analysis Commons](#)

---

## Recommended Citation

Chung, Ming-Hua, "Probabilistic Graphical Modeling on Big Data" (2015). *Theses and Dissertations*. 1415.  
<http://scholarworks.uark.edu/etd/1415>

This Dissertation is brought to you for free and open access by ScholarWorks@UARK. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of ScholarWorks@UARK. For more information, please contact [scholar@uark.edu](mailto:scholar@uark.edu).

Probabilistic Graphical Modeling on Big Data

A dissertation submitted in fulfillment  
of the requirements for the degree of  
Doctor of Philosophy in Mathematics

by

Ming-Hua Chung  
National Central University  
Bachelor of Science in Mathematics, 2004  
University of Arkansas  
Master of Science in Statistics, 2009

December 2015  
University of Arkansas

This dissertation is approved for recommendation to the Graduate Council.

---

Dr. Giovanni Petris  
Dissertation Director

---

Dr. Xiaowei Xu  
Committee Member

---

Dr. Mark Arnold  
Committee Member

---

Dr. Avishek Chakraborty  
Committee Member

## **Abstract**

The rise of Big Data in recent years brings many challenges to modern statistical analysis and modeling. In toxicogenomics, the advancement of high-throughput screening technologies facilitates the generation of massive amount of biological data, a big data phenomena in biomedical science. Yet, researchers still heavily rely on key word search and/or literature review to navigate the databases and analyses are often done in rather small-scale. As a result, the rich information of a database has not been fully utilized, particularly for the information embedded in the interactive nature between data points that are largely ignored and buried. For the past 10 years, probabilistic topic modeling has been recognized as an effective machine learning algorithm to annotate the hidden thematic structure of massive collection of documents. The analogy between text corpus and large-scale genomic data enables the application of text mining tools, like probabilistic topic models, to explore hidden patterns of genomic data and to the extension of altered biological functions. In this study, we developed a generalized probabilistic topic model to analyze a toxicogenomics data set that consists of a large number of gene expression data from the rat livers treated with drugs in multiple dose and time-points. We discovered the hidden patterns in gene expression associated with the effect of doses and time-points of treatment. Finally, we illustrated the ability of our model to identify the evidence of potential reduction of animal use.

In online social network, social network services have hundreds of millions, sometimes even billions, of monthly active users. These complex and vast social networks are tremendous resources for understanding the human interactions. Especially, characterizing the strength of social interactions becomes essential task for researching or marketing social networks. Instead of traditional dichotomy of strong and weak tie assumption, we believe that there are more types of social ties than just two. We use cosine similarity to measure the strength of the social ties and apply incremental Dirichlet process Gaussian mixture model to group tie into different clusters of ties. Comparing to other methods, our approach generates superior accuracy in classification on data with ground truth. The incremental algorithm

also allow data to be added or deleted in a dynamic social network with minimal computer cost. In addition, it has been shown that the network constraints of individuals can be used to predict ones' career successes. Under our multiple type of ties assumption, individuals are profiled based on their surrounding relationships. We demonstrate that network profile of a individual is directly linked to social significance in real world.

## **Acknowledgments**

I would like to specially thank Dr. Giovanni Petris, Dr. Mark Arnold, and Dr. Avishek Chakraborty for supporting me even though I am in special situation, especially Dr. Petris. Without you, I might not be able to start it at the beginning.

I would like specially acknowledge the guidance and support I got from Dr. Xiaowei Xu. He went above and beyond to help me reach where I am. You provide the opportunity that I can only dream of. I truly appreciate it. Thank you, Dr. Xu.

I would also like to specially thank our department staff Mary Powers and Dorine Bower. I will always remember their kindness and support on many aspects of my academic career.

I am grateful to the National Center for Toxicological Research (NCTR) of U. S. Food and Drug Administration (FDA) for internship opportunity through Oak Ridge Institute for Science and Education (ORISE), especially the support from Dr. Weida Tong, Dr. Yuping Wang, Dr. Ge, and Dr. Ayako Suzuki. I would like to acknowledge Binsheng Gong for his assist on data manipulation and functional annotation. I also would like to acknowledge Roger Perkins for providing insightful comments. Gang Chen and Weizhong Zhao also help me on some of the data preparations.

## **Dedication**

This dissertation work is dedicated to my wife, Liya, who has been a constant source of support and encouragement during the challenges of graduate school and life. I am truly thankful for having you in my life. This work is also dedicated to my mom, for her unconditional love and support. You are the best mom a son can have. I dedicate this work to my daughter, who can always give me smile after a hard day's work. This work is dedicated to my parents-in-laws for their selfless support and constant prayers. I really appreciate your help through this challenging time.

I also dedicate this dissertation to my dad, other family members, friends, and church for their support and prayers.

## Table of Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	The Rise of Big Data . . . . .	1
1.2	Probabilistic Graphical Model . . . . .	2
1.3	Probabilistic Topic Models . . . . .	3
<b>2</b>	<b>Asymmetric Author-topic Model for Knowledge Discovering of Big Data in Toxicogenomics [14]</b>	<b>6</b>
2.1	Background and Relative Works . . . . .	6
2.2	Topic Modeling on Microarray Data . . . . .	8
<b>3</b>	<b>Identifying Latent Biological Pathways in Toxicogenomic Data</b>	<b>12</b>
3.1	Dataset . . . . .	12
3.2	Data Preprocessing . . . . .	12
3.3	Model Selection . . . . .	13
3.4	Results . . . . .	14
<b>4</b>	<b>Incremental Dirichlet Process Gaussian Mixture Model on Online Social Networks</b>	<b>19</b>
4.1	Background . . . . .	19
4.2	Related Work . . . . .	22
4.3	Dirichlet Process Gaussian Mixture Model . . . . .	25
4.4	Incremental Learning of Dirichlet Process Gaussian Mixture Model . . . . .	30
4.5	Cosine Similarity . . . . .	36
4.6	Complexity Analysis . . . . .	37
<b>5</b>	<b>Discovering Multiple Social Ties for Characterization of Individuals in Online Social Networks</b>	<b>38</b>
5.1	Datasets . . . . .	38

5.2	Reference algorithms . . . . .	40
5.3	Cluster Assignments . . . . .	40
5.4	Evaluation criteria . . . . .	41
5.5	Results . . . . .	42
<b>6</b>	<b>Discussion and Future Work</b>	<b>57</b>



# 1 Introduction

## 1.1 The Rise of Big Data

Moore's law [38] in 1965 not only predicted the tremendous improvement for semiconductor component technology but also served as a good indicator of how fast the whole computer hardware industry has grown through the decades. Computer hardware in general gets a lot faster, smaller, cheaper, and more powerful. As a result, the rise of "Big Data" becomes inevitable and ubiquitous. In 2001, Doug Laney [31] coined three characteristics which are often used to describe big data over the years: volume, velocity, and variety. That is, besides the size of data sets (volume), the speed of acquisition and processing data sets (velocity) and the various kinds of data sources and structures (variety) are also parts of the big data problem. Beyer and Laney again defined Big Data in 2012 [6] as the following: "Big Data is high volume, high velocity, and/or high variety information assets that require new forms of processing to enable enhanced decision making, insight discovery and process optimization." There are many aspects of tasks involving big data; for example, database warehouse management, data pre-processing, and data modeling, etc. Due to the complex nature of the big data, many traditional statistical or mathematical methodologies simply won't work or are very insufficient to handle the big data problem. Consequently, interdisciplinary subfields (e.g., data mining and machine learning) are created to bridge the gap between big data and the state-of-the-art methodologies. While some area, like text documents or computer images, enjoy the benefits of early success of machine learning algorithms, many areas still rely on traditional algorithms, which are getting more and more insufficient day by day. There are still plenty of areas that haven't benefited from the latest machine learning algorithms.

## 1.2 Probabilistic Graphical Model

As Koller and Friedman defined in their book [30], probabilistic graphical models “use a graph-based representation as the basis for compactly encoding a complex distribution over a high-dimensional space.” Specifically, we are interested in Bayesian network which represents its conditional independence in directed acyclic graphs.

In a traditional graphical model of a Bayesian network:

- Circles represent variables. Specifically, a shaded circle indicates an observed variable and an empty circle indicates an unknown variable.
- Arrow represent conditional dependencies.
- Plate notion indicates the repetition of a relationship for a number of times.

One of the most important features of graphical model is using the combinations of circles and arrows to demonstrate the conditional dependency in a Bayesian network. Consider a Bayesian network in Figure 1 (A) as an example, we can see that variable  $C$  has a set of parent variables,  $A$  and  $B$ , and a offspring variable  $D$ . Based on Bayes’ rule, the joint probability distribution can be written as following:

$$p(A, B, C, D) = P(A)p(B|A)p(C|A, B)p(D|A, B, C) \quad (1)$$

According to the conditional dependency implied in Figure 1 (A), we can simplify the notation of the joint distribution of our model:

$$p(A, B, C, D) = p(A)p(B|A)p(C|A, B)p(D|A, B, C) \quad (2)$$

$$= p(A)p(B)p(C|A, B)p(D|C) \quad (3)$$

Here, Figure 1 (B) shows the graphical representation of a Gaussian mixture model which specified by the following:

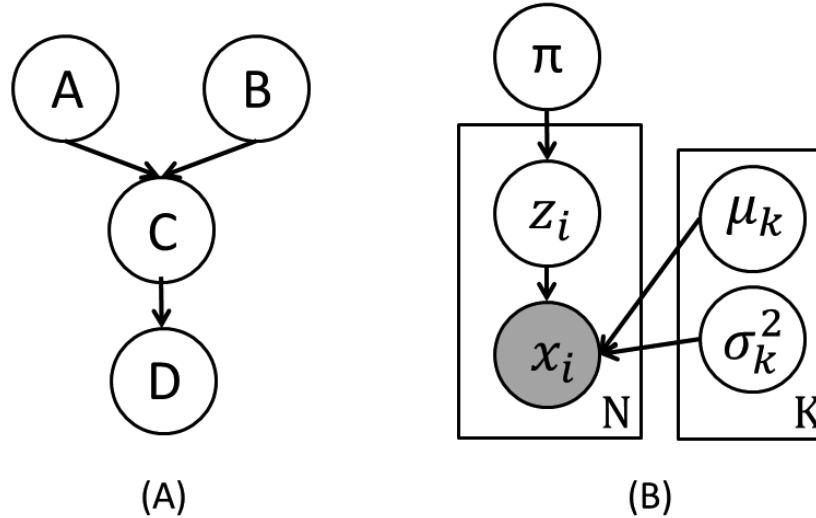


Figure 1: A graphical representation of Gaussian mixture model

- $\pi$  is a  $K$ -simplex which  $\sum_{k=1}^K \pi_k = 1$
- $\forall i = 1, \dots, N,$ 
  - $z_i \in \{1, \dots, K\}$  is the assignments of mixture components.
  - Given  $z_i = k, x_i \sim N(\mu_k, \sigma_k^2)$ .

Therefore, these graphical models not only provide compact visualizations of a complicated distributions, but also help us to understand the conditional dependencies among variables. Besides the two models shown here, well-known models like hidden Markov models or neural networks are all parts of the graphical model family.

### 1.3 Probabilistic Topic Models

The Big Data era also brings digitization of information in all kinds of forms—texts, images, sounds, videos, and social networks. On one hand, the internet along with digitization gives us boundless access to online information to read, to watch, and to listen. On the other hand, it is increasingly difficult to find the information which is relevant to what we are interested in. Over the past decades, the combination of accelerating computer technology and the rise

of big data creates new interests on solving the problem by unsupervised machine learning algorithms.

In 2003, David Blei et al. introduced Latent Dirichlet allocation (LDA)[11], which is among one of the earliest as well as the most important probabilistic topic models. In Blei’s introductory article of the probabilistic topic models, Blei [10] define that “topic models are algorithms for discovering the main themes that pervade a large and otherwise unstructured collection of documents. Topic models can organize the collection according to the discovered themes.” Therefore, finding meaningful “topic” in a large text corpus is the main goal of topic modeling. Furthermore, the probabilistic topic model generally can be seen as a special category of probabilistic graphical models. Therefore, almost all probabilistic topic models can be expressed in a graphical model form. In particular, many probabilistic topic models also assume certain generative process of their observations. Documents are assumed to be generated based on a random mixture of hidden topics, where each topic is a random distribution over a fixed vocabulary of words.

Assume there are  $D$  text documents and each document has  $N_d$  words, where  $d \in \{1, \dots, D\}$ . LDA then follows the generative process below (also see Table 2):

Choose  $\phi_k \stackrel{i.i.d.}{\sim} Dir(\beta)$ , where  $k = 1, \dots, K$ ,  $\phi = \{\phi_1, \dots, \phi_K\}$ .

For each document  $d$ ,

1.  $\theta \sim Dir(\alpha)$ .
2. For each of the  $N_d$  words,
  - (a) choose topic assignment  $z_i \sim Multinomial(\theta)$ .
  - (b) choose a word  $w_n$  from  $p(w_n | z_i = k, \phi) = Multinomial(\phi_k)$ .

Under this assumption, words are organized into topics and each document is controlled by topics. Consequently, instead of dealing with a huge amount of unstructured documents, we are able to browse and interact with these documents through organized “topics”, whose size is often much smaller and hence it is easier to deal with.

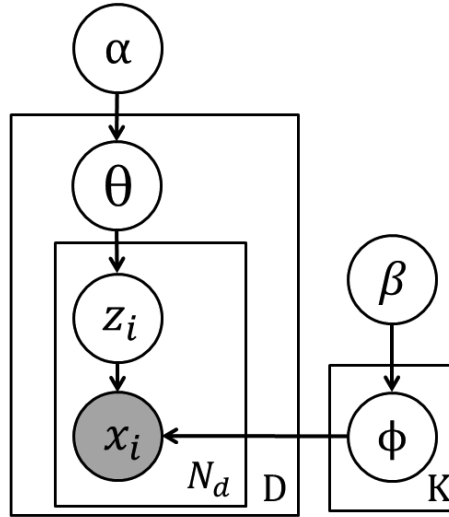


Figure 2: A graphical representation of latent Dirichlet allocation

As Blei point out in his review article of probabilistic topic modeling [10], LDA model can be utilized as a module to be built on. There have been many extensions to the traditional LDA model to accommodate various aspects of big data. In chapter 2, we use a close relative of LDA—author-topic model[45] with some alternations—to explore the hidden patterns in toxicogenomic dataset. In chapter 4, we apply an incremental version of Dirichlet Gaussian Mixture model[33] on social networks to discover multiple types of social ties. At first glance, a Gaussian mixture model may seems to have little connection to LDA. One deals with words—a discrete variable, and another handles numbers—a continuous variable. However, a Dirichlet process version of LDA not only is structurally similar to Dirichlet Gaussian Mixture model, many aspects of our approach to analyze social network data are also influenced by probabilistic topic models. Namely, the characterization of individuals closely based on social ties closely resemble profiling documents based topics. As a document is defined by topic, a person may be defined by his/her social relationships.

## 2 Asymmetric Author-topic Model for Knowledge Discovering of Big Data in Toxicogenomics [14]

### 2.1 Background and Relative Works

As first introduced in 1999, toxicogenomics has emerged as a new subdiscipline of toxicology to take advantage of the newly available genomics profiling technique to gain an enhanced understanding of toxicity at the molecular level [47, 16, 40]. Since then, toxicogenomics significantly contributes to toxicological research and has provided an avenue for joining of multidisciplinary sciences including engineering and informatics into traditional toxicological research [1]. On the other hand, due to high computational cost and lack of advanced knowledge discovery as well as data mining tools, the pace of toxicogenomics has been tardy in recent years [13]. First, a significant deterrent has been the enormous size of toxicogenomic datasets. With perhaps thousands of samples and tens of thousands of genes, the tremendous size of the toxicogenomic database often is cumbersome to handle, analyze and interpret. Gene selection (i.e., selecting relevant genes) and grouping genes (i.e., dealing only partial data at a time) has often been used to reduce complexity and make analyses more tractable [44]. However, both gene selection and grouping run the risk of losing valuable information contained in excluded data. Hence, a method that can efficiently handle the entire data without losing potentially valuable information is desirable. Second, any given biological phenomenon normally involves multiple biological pathways and mechanisms. Currently, some existing clustering algorithms like hierarchical cluster analysis and k-means only allow individuals to be assigned into mutually exclusive clusters. To capture the reality of biological phenomena in gene expression data, we need an algorithm to assign individuals into multiple clusters and to give each cluster a summary of most important genes. One might argue that some fuzzy clustering algorithms [42, 19] are able to assign multiple clusters, yet very few existing algorithm provide much interpretability for clusters. In order to thoroughly utilize the rich interaction in a large database, we desire to organize our samples into meaningful

clusters which can be directly linked by actual biological pathways.

The introduction of *Latent Dirichlet Allocation* (LDA) [11] along with its predecessor *Probabilistic Latent Semantic Analysis* [23] provide a new type of statistical models, namely, probabilistic topic models that have become a standard approach to analyze large collections of unstructured text documents. For a large corpus, probabilistic topic models assume the existence of latent variables (i.e., topics) that govern the likelihood of appearance for each word. Topics are defined as distributions over a fixed vocabulary. Based on the most likely words in each topic, we are able to interpret the meanings of topics. This intuition can be seamlessly transformed into genomics datasets. For a large toxicogenomic data, we assume that there exist latent biological processes that govern alteration of gene expression levels after samples are treated with drugs at various dose levels and time-points. Each latent biological process is characterized by a distribution of a fixed number of genes. By annotating the mostly likely differentially expressed genes in a latent biological process, we then can link the latent variable with a real biological pathway. In recent years, probabilistic topic models have spawned many similar works on genomic data, noticeably in population genetics [43], chemogenomic profiling [18] and microarray data [44, 8, 60]. However, most of the previous works of probabilistic topic models on microarray data either have limited size of samples, or probabilistic topic models are used merely for their clustering ability. The versatility of probabilistic topic models has not been fully assessed. We proposed a probabilistic topic model that was tailored to the structure of a dataset and applied the model to a large toxicogenomics database recently made publicly available. This so-called asymmetric author-topic model (AAT model) combines author-topic model [45] with asymmetric prior [55]. In chapter 2.2, we outlined our data, the proposed model and its application to toxicogenomic data. In chapter 3, we presented the analysis results. Analyses were done with MALLETT [36] that contains the option for asymmetric prior distributions.

## 2.2 Topic Modeling on Microarray Data

### 2.2.1 Latent Dirichlet Allocation on Microarray Data

The fundamental concept of probabilistic topic modeling is the assumption of the existence of latent variables. In *Latent Dirichlet Allocation* (LDA) [11], the latent variables are referred as “topic” and words in documents are chosen based on what topics the document are related to. “Topics” then stands for groups of words that are likely to co-occur in a document. Similar to the previous studies [7, 60], we referred latent variables in toxicogenomics as “latent biological process” and words in documents were replaced by genes. The elements of document-word matrix, which usually are frequencies of occurrences of words in text mining, were transformed to the fold change values in our treatment-gene matrix. Hence, the latent biological processes represent the groups of genes that are significantly co-expressed (or often have high fold change values within groups.). Unlike [44] which alters the original assumption of LDA model, we utilized the original assumption of LDA and this enabled us to implement our models via existing resources of LDA (i.e., MALLET, the open-source software used in our analysis). Therefore, similar to LDA, the model inferences were primarily focused on two probability distributions. In the context of TG-GATEs data, the probability distribution of latent biological processes for each treatment is  $P(Z|Tr)$ , where  $Z$  is defined as latent process assignment while  $Tr$  is defined as treatment to describe biological processes that are activated in a specific treatment. Meanwhile, the probability distribution of gene for each latent biological process is  $P(Ge|Z)$ , where  $Ge$  is defined as genes that are differentially expressed genes (DEGs) from which we are able to associate the latent process to biological pathways. The ability of linking latent process to biological pathway is a definite advantage over other clustering algorithms and we explored its applications in chapter 2.2.3.



Table 1: Summary of different feature specifications of asymmetric author-topic model.

Dataset	Feature	Number of of individuals	Outputs
1	Treatment	1554	$P(Ge Z), P(Z Tr)$
2	Drug	131	$P(Ge Z), P(Z Dr)$
3	Time-dose	12	$P(Ge Z), P(Z DoTi)$

### 2.2.2 Asymmetric author-topic model

Although LDA could be used for treatment-centric analysis, it doesn't take many unique features of the TG-GATEs data into account. In addition to examine the treatment-centric view, drug-centric and/or time-dose-centric analysis were another important component of this study. The author-topic model [45] is a proper methodology to incorporate other aspects of data into model construction. Authorship in author-topic model can be seen as a regrouping of all the documents. While both models are essentially identical, author-topic model groups documents together and give LDA model an author-oriented view for inferences. In other words, once the regrouping is done, the whole process can be seen as an LDA model again. For TG-GATEs data, treatment is defined as a unique drug-time-dose combination, thus we can regroup treatments based on their drug or time-dose to provide a drug-centric or a time-dose-wise analysis. The inferences on models are the same except treatment is replaced by either drug or time-dose. Furthermore,  $P(Z|Tr)$  is replaced by  $P(Z|Dr)$  ( $Dr$  stands for Drug) and  $P(Z|DoTi)$  ( $DoTi$  stands for time-dose) respectively. Table 1 summarizes the total number of individuals in each setting.

As Wallach et al.[55] pointed out, asymmetric prior on the probability distribution of topic for a document substantially increases the robustness of LDA, yet only adds negligible model complexity and computational cost. Therefore, we further improved author-topic model by introducing an asymmetric prior. Here, assume there are  $T$  treatments and each treatment has  $N_t$  genes outcomes, where  $t \in \{1, \dots, T\}$ . Asymmetric author-topic (AAT) model then follows the generative process below:

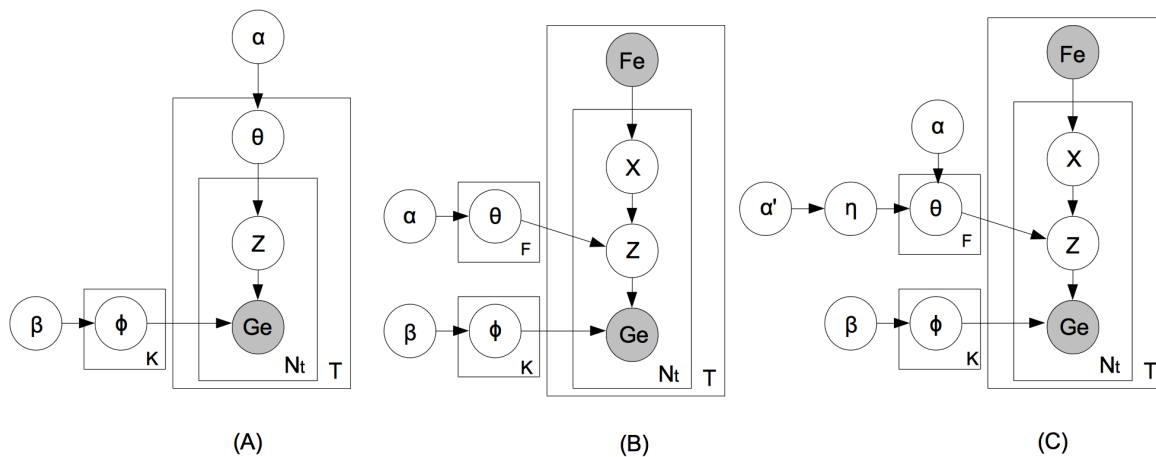


Figure 3: A graphical representation of latent Dirichlet allocation

- Choose  $\phi_k \stackrel{i.i.d.}{\sim} Dir(\beta)$ , where  $k = 1, \dots, K$ ,  $\phi = \{\phi_1, \dots, \phi_K\}$ . Choose  $\eta \sim Dir(\alpha')$
- For each treatment  $t$ , a known value  $Fe_t = f$  is observed, and group assignment  $x_t = f$ ,  $f \in \{1, \dots, F\}$ . Hence, every treatment is assigned into one of the  $F$  feature groups.
  1.  $\theta \sim Dir(\alpha\eta)$ .
  2. For each of the  $N_s$  genes  $Ge_i$ ,
    - (a) choose latent biological pathway assignment  $z_i \sim Multinomial(\theta)$ .
    - (b) choose a gene  $Ge_i$  from  $p(Ge_i|z_i = k, \phi) = Multinomial(\phi_k)$ .

In particular, we can see that only half of the AAT model is different from LDA. First, treatments are regrouped into feature group. In Table 1, we can see that Time-dose has the smallest treatment group, while the first treatment group is essentially assigned to itself and is mathematically equivalent as running a traditional LDA. Second,  $\theta$  now has a hierarchical Dirichlet prior where  $\eta \sim DIR(\alpha')$  and  $\theta \sim Dir(\alpha\eta)$ . If  $\eta$  becomes a unit vector, then the prior becomes symmetric again. Namely, LDA can be seen as a special case of AAT model.

Table 3 shows a comparison of three probabilistic topic models: (A) LDA, (B) Author-topic model, and (C) Asymmetric author-topic model.

The asymmetry of priors can be easily achieved since the chosen software MALLET has a build-in option in the command. More information about MALLET can be found on their website (<http://mallet.cs.umass.edu/>).

### 2.2.3 Functional Annotation and Similarity Ranking

One essential aspect of any clustering algorithm is to organize individuals into their respective clusters. However, the clusters often are difficult to interpret. Through AAT model, individuals are clustered to multiple latent biological processes based on the probability distribution  $P(Z|Tr)$  (or  $P(Z|Dr)$ ,  $P(Z|DoTi)$ ). For each latent biological process, probability distribution  $P(Ge|Z)$  controls how likely each gene is differentially expressed. According to our results, there are often fewer than 200 genes (out of 31,042 total genes) that have positive probability in each latent biological process while other genes have probability of zeros. We then annotate the found list of DEGs in each latent biological process through online database DAVID [24]. Consequently, every feature (i.e., treatment, drug, or time-dose) in the database is automatically connected to annotated biological pathways. The ability of our proposed model to link from the latent biological processes to functional annotation, such as real biological pathways, is a significant advantage over other existing methods. Another application of author-topic model is to find the feature most similar to a given one. We can quantitatively measure the similarity between a pair of features by calculating the symmetric Kullback–Leibler divergence (sKL) [45] between a pair of  $P(Z|Tr)$  (or  $P(Z|Dr)$ ,  $P(Z|DoTi)$ ). For instance, by finding the sKL between  $P(Z|Dr_1)$  and  $P(Z|Dr_2)$ , we can tell how similar Drug 1 and Drug 2 is (i.e., a low sKL score indicates that two drugs exhibit similar topic distributions.). Given a drug, our model is able to recommend a list of drugs ranked by the similarity score sKL. Due to (1) the similarity is based on  $P(Z|Dr)$ , the probability of latent biological processes given drugs, and (2) all the latent biological processes

are able to annotated to biological pathways, we know which drugs are similar as well as exactly which pathways link them together.

### 3 Identifying Latent Biological Pathways in Toxicogenomic Data

#### 3.1 Dataset

The Japanese Toxicogenomics Project [54, 13] is a 10-year collaborative project involving two Japanese government institutes and 18 private companies [25]. The project produced a comprehensive gene expression database, called Open TG-GATEs for the effects of 170 compounds (drugs) on liver and kidney as primary target organs in both *in vivo* and *in vitro* experiments. Specifically, in the *in vivo* experiment, animals are treated at three different doses (low, middle, and high) of drugs once every day for four different treatment durations (3, 7, 14, and 28 days). In addition, control animals are concurrent with all the twelve combinations of doses and durations. More details on the animals and experimental design have been described previously [53]. Microarray based gene expression data were generated using the  $\text{\textcircled{R}}$ GeneChip Rat Genome 230 2.0 Arrays (Affymetrix, Santa Clara, CA, USA) that contains 31,042 probe sets. The data used in this study is obtained from the Annual International Conference on Critical Assessment of Massive Data Analysis (CAMDA) 2013 ([http://dokuwiki.bioinf.jku.at/doku.php/tgp\\_prepro](http://dokuwiki.bioinf.jku.at/doku.php/tgp_prepro), accessed on April 8th, 2014). In this study, only the data from *in vivo* repeated dose experiment was used.

#### 3.2 Data Preprocessing

Similar to others [44, 7, 60], our first step of analysis was to obtain a “document-word” matrix for gene expression data to apply topic model. Instead of the sample-gene expression matrix used in others’ works, we created treatment-fold change matrix for our studies. This was due to the fact that TG-GATEs has multiple treated samples for one treatment (a unique drug-time-dose combination) along with controlled group. Therefore, we were able

to apply a more refined treatment-fold change matrix as our inputs. Here, all fold change values of gene expressions between treated and control samples were calculated and used as the value of elements of the matrix. Genes with absolute fold change greater than 1.5 were considered as differentially expressed genes (DEGs) and set the fold change values zeros for the non-DEG. The final product is a treatment-fold change matrix where each column represents a treatment and each row represents a gene.

### 3.3 Model Selection

We run all three of our models on MALLET, whose model inference is based on Gibbs sampling algorithm. One common concern using Gibbs sampling is the convergence of the model. Generally, convergence of the model is monitored via tracking the probability of the likelihood function after burn-in. After the likelihood probability stabilizes, we can deem convergence to be adequate. We run 3000 iterations for all models and observe stability after about 1,500 iterations. We also perform sensitivity analyses for major parameters, including number of latent biological processes, and the initial values for hyperpriors. Hyperpriors are usually not big factors in the model as they are constantly revised during rounds of Gibbs sampling inference. On the other hand, the number of latent biological processes is important. While there is no way to know how many biological processes are involved in the whole database, we can estimate the number based on perplexity performance [11]. In addition, asymmetric topic models have been shown to be robust to variations in the number of topics [55]. All the parameters are chosen based on 10-fold cross-validation. For model 1 (treatment), the number of latent biological processes is 200. For model 2 and 3 (drug and time-dose) the number of latent biological processes is 100.

## 3.4 Results

### 3.4.1 AAT model on Glutathione Depletion

One proven application of TGP database is detection of glutathione depletion [54]. Taking well-known hepatotoxin acetaminophen as an example, it was reported that glutathione metabolism was related to acetaminophen-induced hepatotoxicity and the mechanisms that underline such liver injury [2, 5]. For instance, James et al.[27] pointed out that acetaminophen could induce potentially fatal, hepatic centrilobular necrosis when taken in overdose, since the amount of active metabolite overwhelmed the detoxification capacity of intracellular glutathione. Among our proposed models, model 1 gives us a treatment-centric view of the TGP database. Table 2 shows  $P(Z|Tr)$  from model 1 that represents the most likely latent biological processes that encode biological phenomena associated with acetaminophen. Here, only top three topics for each different treatment (drug-dose-time) are shown (for full table, see Supplementary 1). Latent process 161 is identified in 8 out of 12 time-dose combinations for acetaminophen, as early as the three-day treatment with the middle dose of 600 mg. Furthermore, the list of most probable DEGs for latent process 161 is extracted from  $P(Ge|Z)$  and functionally annotated by online database DAVID. In Table 3, functional annotation is done on online database David. Only the top 3 annotated of Kyoto Encyclopedia of Genes and Genomes (KEGG) pathway terms are shown here (for full table, see Supplementary 2). As seen on Table 3, glutathione metabolism pathway is significantly identified in the KEGG database, which is consistent with the previous findings.

In model 2, the drug-centric view of the TGP database, we observe similar results. Again, the most likely active latent process for acetaminophen is latent process 92 (Table 4) and it is once again significantly identified as glutathione metabolism pathway in the KEGG database (Table 5). Again, only top three latent processes for each drug are shown (For full table, see Supplementary 3 and 4 respectively). In addition, by simply searching the drugs that also have No. 92 among the top ranked latent processes, we find that bromoben-

Table 2: The probability of latent biological processes for acetaminophen under model 1.

Treatment index	Dose	Time (Days)	Top ranked Latent Biological Processes					
			1	Probability	2	Probability	3	Probability
36	Low	3	2	0.149	36	0.124	181	0.122
37	Middle	3	161	0.279	111	0.168	116	0.098
38	High	3	161	0.139	39	0.1	169	0.1
39	Low	7	68	0.305	162	0.211	69	0.165
40	Middle	7	161	0.366	149	0.12	57	0.079
41	High	7	161	0.275	27	0.08	39	0.066
42	Low	14	69	0.153	134	0.138	63	0.138
43	Middle	14	161	0.342	128	0.104	37	0.098
44	High	14	161	0.274	113	0.082	128	0.074
45	Low	28	69	0.175	96	0.175	160	0.153
46	Middle	28	161	0.278	96	0.152	14	0.085
47	High	28	161	0.366	197	0.091	164	0.07

Table 3: Functional annotation of KEGG pathways on latent biological process 161 under model 1.

Term	Count	FDR	P-value	Gene
rno00480:Glutathione metabolism	8	1.55E-05	1.65E-08	GPX2, GSR, GCLC, G6PD, GSTA5, GCLM, GSTP1, MGST2
rno00980:Metabolism of xenobiotics by cytochrome P450	7	0.00142	1.51E-06	GSTA5, ADH4, UGT2B1, EPHX1, CYP3A9, GSTP1, MGST2
rno00982:Drug metabolism	7	0.00420	4.47E-06	GSTA5, ADH4, UGT2B1, AOX1, CYP3A9, GSTP1, MGST2

Table 4: The probability of latent biological processes for acetaminophen, bromobenzene, chlormezanone, coumarin, methimazole, and ticlopidine under model 2.

Drug index	Dose	Top ranked Latent Biological Processes					
		1	Probability	2	Probability	3	Probability
3	acetaminophen	<b>92</b>	0.201	17	0.190	1	0.118
16	bromobenzene	<b>92</b>	0.318	1	0.138	17	0.125
27	chlormezanone	9	0.341	<b>92</b>	0.192	1	0.128
37	coumarin	98	0.293	<b>92</b>	0.193	1	0.142
81	methimazole	<b>92</b>	0.211	21	0.185	32	0.143
123	ticlopidine	9	0.248	<b>92</b>	0.093	1	0.089

Table 5: Functional annotation of KEGG pathways on latent biological process 92 under model 2.

Term	Count	FDR	P-value	Gene
rno00480:Glutathione metabolism	11	5.67E-07	5.18E-10	GSTM1, GPX2, GSR, GCLC, GSTM4, G6PD, GSTA5, GSTT1, GCLM, GSTP1, GSTM7, MGST2
rno00980:Metabolism of xenobiotics by cytochrome P450	9	0.00384	3.51E-06	GSTM1, GSTM4, GSTA5, ADH4, UGT2B1, EPHX1, GSTT1, GSTP1, GSTM7, MGST2
rno00982:Drug metabolism	9	0.00420	4.47E-06	GSTM1, GSTM4, GSTA5, ADH4, UGT2B1, AOX1, GSTT1, GSTP1, GSTM7, MGST2

zene, chlormezanone, coumarin, methimazole, and ticlopidine strongly link with glutathione metabolism pathway (Table 4), and hence presumably become causes of glutathione depletion. Such hepatotoxicity associated with these 6 drugs through the glutathione metabolism pathway is well supported in other studies (Jollow et al., 1974;Thor et al., 1979;Wright et al., 1996;Mizutani et al., 2000;Uehara et al., 2010;Shimizu et al., 2011). Overall, our results indicate that the construction of our proposed model indeed matches with the well-known biological processes and hence the model is able to detect potential treatments or drugs that cause glutathione depletion.



Table 6: Most similar drugs to acetaminophen based on sKL scores.

Drug name	sKL score
bromobenzene	3.04238
phenacetin	4.47157
bucetin	4.51243
cimetidine	5.46445
disopyramide	5.85482
cephalothin	5.89109
papaverine	5.92761
Erythromycin ethylsuccinate	5.92976
coumarin	6.03134
nitrofurantoin	6.03479

### 3.4.2 AAT model on Drug Similarity and Potential Reduction of Animal Use

Through sKL score (described in chapter 2.2.3), functional similarity of drugs can be explored. As an example, we can obtain the most functionally similar drugs to acetaminophen as shown in Table 6. Here, the smaller the sKL is, the more similar two drugs are. Notice only top 10 ranked drugs are shown here (For full table, see Supplementary 5). The drugs that have smaller sKL score with acetaminophen (i.e., a pair-wise score) will exhibit most similar latent biological processes. We can observe that bromobenzene and coumarin, which linked through glutathione depletion pathway, are on the list.

Another application of sKL score is to be used as potential evidence of reduction of animal use. Reducing, replacing and refining animal use (3Rs) has been increasingly a goal in toxicogenomics [46, 56]. While dose level and time-point are expected to be important, there is generally no easy way to determine which treatment is ignorable for a given drug. sKL scores measure the similarity between a pair of treatments. The idea is to see if either dose or time in treatments of a drug does not play a significant role to affect sKL score. If one of them is not significant to sKL score, then there exists the potential to reduce the number of treatments without compromising study goals. Similar to multivariate analysis of variance (MANOVA), the importance of dose and time can be attained with generalized

linear models on sKL scores as the following:

$$sKL = \beta_1 X_{Dose} + \beta_2 X_{Time}, \quad (4)$$

$$sKL = \beta_1 X_{Dose}, \text{ or} \quad (5)$$

$$sKL = \beta_1 X_{Time}. \quad (6)$$

Here,  $X_{Dose}$  is defined as a categorical variable that includes six different dose pairs (i.e., Low-Low, Low-Middle, Low-High, Middle-Middle, Middle-High, and High-High).  $X_{Time}$  is defined as a continuous non-negative variable that represents the difference between two time-points. By fitting the generalized linear model using various common model criteria (e.g., adjusted R-square, AIC, and BIC), we can compare dose and/or time significance regarding to sKL score. A level of feature that has no significant impact on sKL score can be potentially reduced. While only having 12 individuals, model 3 can be used to detect the overall significance of dose and time. Unsurprisingly, dose and time generally are both significant to sKL score as seen in Table 7. It is naïve to think we can remove any treatment regardless which drug is been tested, yet there might be specific drugs that fit our assumption. As examples, we chose acetaminophen, coumarin, and benzbromarone to be tested in the generalized linear models. Among all, only benzbromarone consistently demonstrate the superiority of dose only model under all three model criteria. Therefore, it is possible to combine time-points for treatments of benzbromarone due to the insignificance of time regarding to sKL score.

Table 7: Generalized linear models for sKL scores under three (Adjusted R-square, AIC, and BIC) criteria, with best outcomes bolded.

GLMs	Adjusted R-square			AIC			BIC		
	D&T	Dose	Time	D&T	Dose	Time	D&T	Dose	Time
Model 3	<b>0.456</b>	0.437	0.076	<b>82.703</b>	93.771	117.212	<b>98.030</b>	106.909	121.591
acetaminophen	<b>0.559</b>	0.453	0.051	<b>204.660</b>	216.462	246.815	<b>219.988</b>	229.600	251.194
coumarin	<b>0.592</b>	0.583	0.016	258.487	<b>257.649</b>	296.490	273.814	<b>270.786</b>	300.869
benzbromarone	0.813	<b>0.816</b>	0.004	225.281	<b>223.221</b>	340.736	240.609	<b>236.359</b>	345.115

## 4 Incremental Dirichlet Process Gaussian Mixture Model on Online Social Networks

### 4.1 Background

Recent explosive growth of online social networks such as Facebook and Twitter provides a unique opportunity for many data mining applications including real time event detection, community structure detection and viral marketing. While many researches focus on characteristics of individuals, we aim at the building blocks of network structure—social ties. As it is said, “It’s not what you know, it’s who you know.”

In his 2004 article [12], renowned social network scientist Ronald Burt demonstrates that the network constraints of a person’s social network can be used to predict one’s career success (e.g., salary, evaluation, or performance). In other words, a person with open network around (i.e., surrounded by weak ties) has better chance to become successful comparing to a person with closed network (i.e., surrounded by strong ties). Therefore, by simply analyzing individual’s surrounding network, we will not only be able to chart their importance regarding to the whole network, but also link them into real life performances.

There have been various studies which aim to understand the essence of social ties in sociology and computational sciences [22, 17, 41]. However, studies often measure the resemblance between two persons by user profiles. Similar to [49], we choose to measure the tie strength by merely using the graph structure in the social networks. In particular, each social

tie has a tie strength, which can be estimated by a ratio of neighborhood overlap between two adjacent vertices of the edge [17, 41]. Among many measure of the strength of social tie (e.g., Jaccard index, cosine similarity, and topological overlap matrix [32]), we choose cosine similarity since: (1) geometric mean (i.e., cosine) is generally stabler than arithmetic mean (i.e., Jaccard), and (2) cosine [58].

In the past, social tie studies heavily relied on the assumption that there existing merely two types of ties—strong and weak—in a static social network. Social relationships are very complex and can consist of different kinds of ties including strong ties (e.g., close friends, family members), weak ties (e.g., acquaintances), or something in between (e.g., colleagues, co-authors, Twitter followers, etc.). We believe simple dichotomy is too generalized. Social relationship are very complex and can consist of different kinds of ties including strong ties (e.g., close friends, family members), weak ties (e.g., acquaintances), or something in between (e.g., colleagues, co-authors, Twitter followers, etc.). Imaging a scenario shown in Figure 4. Some ties (e.g., the solid line in Figure 4) form and bind community structures, while each may be knitted with a different density. Some ties (e.g., short dash line between  $D$  and  $E$  in Figure 4) serves as the bridge between different community structures. Finally, some ties (e.g. long dash lines between  $C$  and  $R$ , and between  $C$  and  $Q$  in Figure 4) connect with individuals who are not members of any community. Here, a hub like  $R$  plays a special role which connect multiple communities, while outliers like  $Q$  and  $S$  are individuals on the margin of community structures. To properly classify ties in this scenario, a simple dichotomy between strong and weak ties will not be enough. Under our current highly interconnected society, we aim to develop a framework that can accommodate the real complexity of social networks.

Besides multiple types social ties, another crucial aspect that are often ignored is the dynamics of social networks. Social ties are dynamic in the sense that a new tie may be established through a meeting; and an existing tie may be either strengthened or weakened due to the change of the proximity. Therefore, one remaining critical challenge of mining

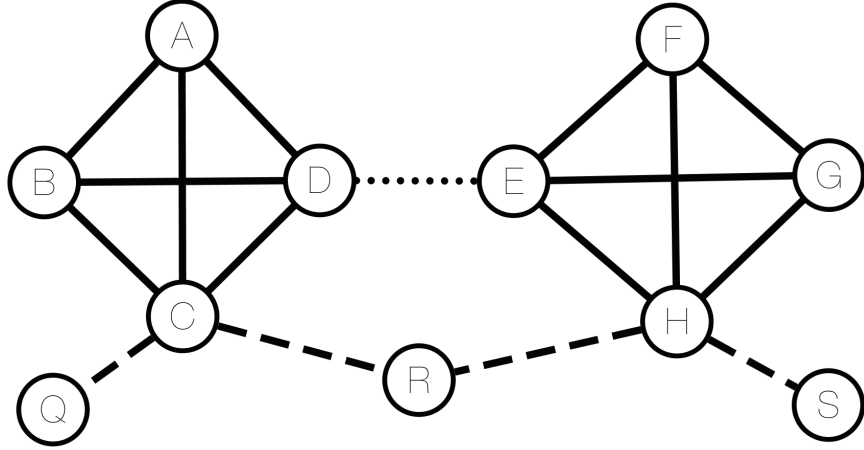


Figure 4: A scenario of multiple types of ties

online social networks is about understanding the dynamic nature of complex online relationship between individuals. To this end, we apply the Dirichlet Process Gaussian Mixture Model (DPGMM) [33] on cosine similarity of social ties. One of the most difficult problems in clustering is determining the total number of components. In contrast to traditional finite mixture models, DPGMM infers the number of components from data by using the Dirichlet process, which let data to determine the number of components to be generated. We further enhance Lin’s DPGMM [33] to an incremental algorithm for dynamic social networks. While an update of a tie (e.g., adding or removing) can cause changes to every adjacent ties, our incremental algorithm requires re-run merely on the data that are affected by the update; that is, it doesn’t require rerun on the whole data. This is especially useful for big data like Facebook or Twitter.

The main contribution of our work is as follows:

1. We lay out the framework to cluster social ties beyond strong and weak ties in order to reflect the true hyper-interconnected nature of social networks. We use real world data to test the ability of our approach to capture multiple types of social ties.
2. We apply an incremental algorithm for dynamic social networks. Our algorithm supports both insertion and deletion as basic operations for any social tie update to online

social networks.

3. The performance of the proposed algorithm is evaluated by the accuracy of identified different types of social ties, as well as the running time using some real social networks. The experiment demonstrates that our algorithm is scalable to large dynamic social networks and can achieve a more accurate result comparing with existing algorithms.
4. We demonstrate that individual network profiles generated from our model can be linked directly to real social significance. We further demonstrate the model ability to measure network constraints of the communities in a online social network.

The study is organized as follows. We first give an overview of related work in chapter 4.2. The proposed Dirichlet process Gaussian mixture model and an incremental model inference algorithm are presented in chapter 4.3. The performance of proposed method is evaluated on real social networks. The experimental design and result are described in chapter 5. Finally we conclude the study with a summary and future works in chapter 6.

## 4.2 Related Work

The study of social ties is a major task in sociology. Granovetter [22] first investigated the functional role of social ties. Granovetter [22] showed that “weak ties” would play critical roles of bridging communities. In his seminal article entitled “*The Strength of Weak Ties*”, the “*weak tie hypothesis*” postulates that individual community structures of a social network are predominately bounded by strong ties, while weak ties function as bridges connecting these densely knit community structures. In theory, a person with many weak ties (i.e., having a open network) tends to become a key role to translate information among different communities and hence is essential to the whole network. On the other hand, a person with many strong ties (i.e., having a closed network) has lesser impact on the community. Removing persons with many strong ties will not affect the structure of the network significantly since their many strong tie neighbors can take their place in holding the communities

together. However, social ties at the time are rather elusive to quantify and strongly suffer from cognitive biases, errors of perception, and ambiguities, especially when the data collection is based on subjective self-reports from participants.

Since then, the rise of online social networks provide new opportunities on social tie studies. Many researchers [29, 21, 20] relied on supervised learning which required labeled training data that may not be available or difficult to obtained. Xiang et al. [57] develop an unsupervised model to estimate tie strength from user interactions (e.g., communication, tagging) and user similarity. In contrast to binary social ties their method can handle various social ties such as close friends and acquaintances. Jones et al. [28] provide a study of relevant features of strong ties and find the frequency of online interaction is diagnostic of strong ties and is more informative than attributes of the user and the user’s friends. Tang et al. [51] develop a semi-supervised learning framework to classify various social ties such as colleagues and intimate friends. More specifically, they use user and link characteristics to build a generative model that assigns the most likely type to a specific relationship. In a follow-up study [50], they further generalize the proposed model for classifying social ties by learning across heterogeneous networks through incorporating ideas from social theories such as structural balance and social status. Although the approaches above either don’t require labeled training data or only a portion of data being labeled, their all need user information such as user profiles that may be noisy or incomplete. Recently Backstrom et al. [3] propose a new network measure, dispersion, for the recognition of romantic partner of Facebook users, which only uses the structure of the Facebook. Dispersion is designed for the identification of romantic relationship, which is only a special type of strong ties; and may not be generalized to the characterization of other social ties.

Recently, Sintos and Tsaparas [49] characterize the social ties into strong or weak ties based on the Strong Triadic Closure (STC) principle. They are also among the first to suggest the existence of multiple (strong) ties (e.g., strong family ties, strong work ties, or strong friendship ties). Between the two algorithm proposed in [49] (i.e., Greedy and Max-

imalMatching algorithms), Greedy algorithm achieves a better performance and produces consistently a larger number of strong edges comparing to the MaximalMatching algorithm. Our approach is closely related to their work in many aspects of the studies, yet our approach overcomes several shortcomings of Greedy algorithm. The differences are summarized as follows:

1. Both develop methods for the characterization of social ties by solely using the network structure. Yet, Sintos and Tsaparas use either count of coexistence or Jaccard similarity as measure of tie strength, while we choose cosine similarity.
2. Whenever new edges are formed (or old edges are removed), non-incremental algorithms require rerun of the whole data—is costly and time-consuming. Incremental algorithms only require rerun for the edges that are affected by the changes. It provides speed and cost advantage over traditional algorithm, especially if the changes of edges are relatively small comparing to the whole data set. While Greedy+ algorithm is able to add new edges iteratively, it does not provide support for edge removal. On the other hand, our model can handle both addition and removal of ties—hence, a true incremental algorithm.
3. Finally, both of our works consider multiple types of social ties. In the MultiGreedy Algorithm of [49], Greedy algorithm is repeatedly reused on the leftover weak ties in order to generate new batch of strong ties. However, there is no natural way to stop the iterative process of MultiGreedy algorithm—the number of types of social ties need to be predetermined. On the contrary, our model is built on Dirichlet process, which allow data themselves to determine the number of components.

The majority of our proposed algorithm has been discussed in Lin’s work [33]. However, we make several improvements. First, we extend the original algorithm to a true incremental one. Second, we derived a simple calculation of log-likelihood for cluster assignments of ties (chapter 5.3). Third, while Lin shows the mathematical superiority of DPGMM in terms



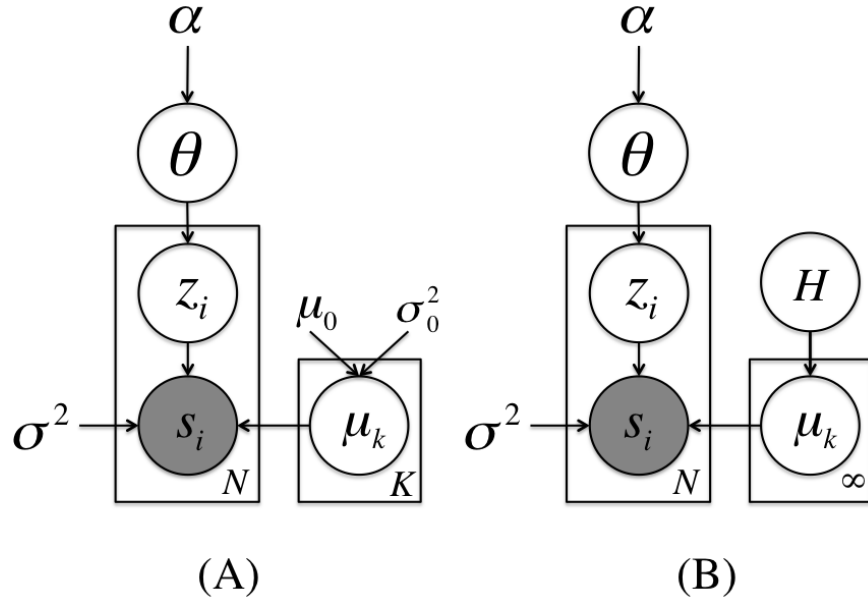


Figure 5: (a)Finite mixture model; (b)Dirichlet process mixture model

of log-likelihood performance, we further demonstrate the speed and classification accuracy advantage of DPGMM in chapter 5.

### 4.3 Dirichlet Process Gaussian Mixture Model

Our hypothesis is that there exists different types of social ties; each type of ties can be characterized by a statistical distribution. In our preliminary investigation, Gaussian distribution performs better than other distribution assumption (e.g., Pareto, Beta). Therefore, we applied mixture of Gaussian distributions on social ties.

A finite Gaussian model (Figure 5(a)) with  $K$  components and fix variance  $\sigma^2$  can be seen as a generative process:

1. Choose  $\boldsymbol{\theta} \sim \text{Dirichlet}(\alpha)$ , where  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_K)$ .
2. Choose  $\boldsymbol{\mu} \sim \text{Gaussian}(\mu_0, \sigma_0)$ , where  $(\mu_0, \sigma_0)$  is the hyperparameter of the component mean  $\boldsymbol{\mu} = (\mu_1, \dots, \mu_K)$ .
3. For each observation  $s_i, \forall i = 1, \dots, N$ ,

- (a) Draw the assignment of component  $p(z_i|s_i) \sim Multi(\boldsymbol{\theta})$ , where  $\mathbf{z} = (z_1, \dots, z_N)$  is the assignment of components for social ties.
- (b) Once we draw  $z_i = k$ , tie strength  $s_i$  is then generated from  $p(s_i|z_i, \mu_k) \sim N(\mu_k, \sigma^2)$ .

The traditional mixture model requires the number of components  $K$  known as a priori. Because of the dynamic nature of online social networks, a fixed number of component may not be flexible enough to adapt the dynamic social networks. On the contrary, Bayesian nonparametric models such as Dirichlet process mixture models (DPMM) allow the number of components to vary during learning, thus providing great flexibility for analysis. DPMM generally can be seen as a generative process with stick-breaking construction(Figure 5(b)). Similar to the Gaussian Mixture model, we can describe a Dirichlet process Gaussian mixture model as follow:

1.  $\forall k = 1, 2, \dots, \infty,$

$$\beta_k \sim Beta(1, \alpha), \quad \theta_k = \beta_k \prod_{l=1}^{K-1} (1 - \beta_l),$$

$$\mu_k \sim H = N(\mu_0, \sigma_0), \quad G = \sum_{k=1}^{\infty} \theta_k \delta_{\mu_k}.$$

2. For each observation  $s_i, i = 1, \dots, N,$

- (a) Draw the assignment of component  $z_i \sim Multi(\boldsymbol{\theta})$ .
- (b) Once we draw  $z_i = k$ , tie strength  $s_i$  is then generated from  $p(s_i|z_i, \mu_k) \sim N(\mu_k, \sigma^2)$ .

As shown by Sethuraman [48], we define a Dirichlet process  $G$  distributed with concentration parameter  $\alpha$  and base distribution  $H$ , denoted by  $G \sim DP(\alpha, H)$ . There are two major differences between these otherwise similar generative processes. First, the prior of  $\boldsymbol{\mu}$ , is changed from a pair of parameters  $(\mu_0, \sigma_0)$  to  $H$ , a Gaussian distribution with mean

$\mu_0$  and variance  $\sigma_0$ . Second, there is no need to fix number of components  $K$  as in finite mixture model since Dirichlet process automatically generated  $K$  from observation  $\mathbf{s}$  via the stick-breaking construction.

### 4.3.1 Variational Inference

Lin developed an algorithm [33] to infer the component parameters  $\boldsymbol{\mu}$  through a predictive distribution of  $\boldsymbol{\mu}$ :

$$p(\boldsymbol{\mu}|\mathbf{s}) = \mathbb{E}_{G|\mathbf{s}}[p(\boldsymbol{\mu}|G)] \quad (7)$$

The expectation is taken through  $p(G|\mathbf{s})$ , and the goal is to find a tractable posterior distribution of  $p(G|\mathbf{s})$  via variational inference. Assume  $N$  samples have been generated by  $G \sim DP(\alpha, H)$  and it contains  $K$  components with component parameter  $\boldsymbol{\theta} = (\theta_1, \dots, \theta_K)$ . Let  $C_1, \dots, C_K$  be the partition corresponding to component assignment  $\mathbf{z}$ . Then, the posterior distribution of  $G$  (denoted by  $\hat{G}$ ) is defined by:  $\hat{G} \sim DP(\hat{\alpha}, \hat{H})$ , with  $\hat{\alpha} = \alpha + N$ , and

$$\hat{H} = \frac{\alpha}{\alpha + N}H + \sum_{k=1}^K \frac{|C_k|}{\alpha + N} \delta_{\mu_k}. \quad (8)$$

This expressions has a straightforward interpretation on how Dirichlet process assigns new individuals into clusters. For an existing cluster  $k$  ( $1 \leq j \leq K$ ) with mean  $\mu_j$ , the probability of assigning a new observation into cluster  $k$  is proportional to the number of observations which are already in the cluster  $k$ ; namely,  $|C_k|$ . On the other hand, the probability of assigning observation into a brand new cluster  $K + 1$  is proportional to the pseudo count  $\alpha$ , and a new mean  $\mu_{K+1}$  is again generated from base distribution  $H$ . The posterior distribution of Dirichlet process  $G$  is approximated by a variational distribution  $q(G|\boldsymbol{\rho}, \boldsymbol{\nu})$ :

$$\begin{aligned} p(G|\mathbf{s}) &= \sum_{\mathbf{z}} p(\mathbf{z}|\mathbf{s})p(G|\mathbf{s}, \mathbf{z}) \approx \sum_{\mathbf{z}} \prod_{i=1}^N \rho(z_i)q_{\nu}(G|\mathbf{z}) \\ &\doteq q(G|\boldsymbol{\rho}, \boldsymbol{\nu}) \end{aligned} \quad (9)$$

Table 8: Symbol tables

---

$K$	Number of components
$N$	Total number of social ties
$\mathbf{s} = (s_1, \dots, z_N)$	Cosine similarities of social ties
$\mathbf{z} = (z_1, \dots, z_N)$	Component assignment for social ties
$G$	Dirichlet process, $G \sim DP(\alpha, H)$
$\alpha$	Concentration parameter of the Dirichlet process $G$
$H$	Base distribution, $H = N(\mu_0, \sigma_0^2)$
$\boldsymbol{\theta} = (\theta_1, \dots, \theta_K)$	Parameters of multinomial distribution that generate $\mathbf{z}$
$C_1, \dots, C_K$	Partitions of $\mathbf{s}$ corresponding to current component assignment $\mathbf{z}$
$\boldsymbol{\mu} = (\mu_1, \dots, \mu_K)$	Means of the mixture Gaussian distributions that generate $\mathbf{s}$ . One for each component $k$ .
$\sigma^2$	Variance of mixture Gaussian distribution that generate $\mathbf{s}$ . Fixed for all components.
$\rho_k^{(i)} = \rho(z_i = k)$	Variational distribution of $p(z_i = k   s_i)$ for each component $k$ at $i$ -th iteration
$\nu_k^{(i)} = \nu_k^{(i)}(\mu_k)$	Variation distribution of point mass of $\mu_k$ for each component $k$ at $i$ -th iteration

---

Notice  $p(\mathbf{z}|\mathbf{s})$  is approximated by variational distribution  $\prod_{i=1}^N \rho(z_i)$ , and  $p(G|\mathbf{s}, \mathbf{z})$  is approximated by variational distribution  $q_\nu(G|\mathbf{z})$ . Here, both  $p(G|\mathbf{s}, \mathbf{z})$  and  $q_\nu(G|\mathbf{z})$  are special cases of Normalized Random Measures with independent increments [26]. According to Lemma 1 of [33], we have

$$\nu_k \propto H(d\boldsymbol{\mu}) \prod_{i \in C_k} p(s_i|\boldsymbol{\mu}), \quad \forall 1 \leq k \leq K, \quad 1 \leq i \leq N. \quad (10)$$

Hence, we can update (8) by replacing  $p(G|\mathbf{s})$  with the tractable variational distribution  $q(G|\boldsymbol{\rho}, \boldsymbol{\nu})$ :

$$\hat{H} = \frac{\alpha}{\alpha + N} H + \sum_{k=1}^K \frac{\sum_{i=1}^n \rho_i(z_i = k)}{\alpha + N} \nu_k. \quad (11)$$

By comparison to (8), we have  $|C_k| = \sum_{i=1}^n \rho_i(z_i = k)$ , which records the current counts of observations in each clusters. In addition,  $\alpha$  still works as a pseudo count for brand new clusters. Consequently, we will have the same interpretation as we do in (8) except we keep tracking the posterior distribution  $\nu_k$  instead of the point mass of  $\mu_k$ .

### 4.3.2 Conjugate prior and exponential family

In general, an exponential family is a set of probability distributions that have a specific format:

$$p(s|\mu) = \exp\{\eta(\mu)T(s) - A(\mu) + B(s)\} \quad (12)$$

Here,  $s$  has a distribution of exponential family with parameter  $\mu$ .  $\eta(\mu)$  is called natural parameter,  $T(s)$  is called sufficient statistics, and  $A(\mu)$  is called log-partition. Assume prior distribution  $H = p(\mu|\lambda, \lambda')$  and consider the probability density function (pdf) has following form:

$$p(\mu|\lambda, \lambda') = \exp\{\lambda\eta(\mu) + \lambda'(-a(\mu)) - A(\lambda, \lambda') + B(\mu)\} \quad (13)$$

Then, regarding to  $\mu$ ,  $(\eta(\mu), (-a(\mu)))$  are the sufficient statistics, and  $(\lambda, \lambda')$  are the natural parameters. In addition, assume the pdf of the observation  $s$  has the following form:

$$p(s|\theta) = \exp\{\eta(\mu)T(s) - \gamma a(\theta) + b(s)\} \quad (14)$$

Similarly,  $T(s)$  is the sufficient statistics, and  $\eta(\mu)$  is the natural parameter regarding to  $s$ .

We claim that  $H$  is the conjugate prior of  $p(s|\mu)$ , if the posterior distribution  $p(\mu|s, \lambda, \lambda')$  has the same type of probability distribution as  $H$ . Applying basic Bayes' theorem, we can derive the the posterior distribution of  $\mu$ :

$$\begin{aligned} p(\theta|s, \lambda, \lambda') &\propto p(\theta|\lambda, \lambda')p(s|\theta) \\ &= \exp\{(\lambda + t(s))\eta(\theta) + (\lambda' + \gamma)(-a(\theta)) \\ &\quad - A(\lambda, \lambda') + B(\theta) + b(s)\}. \end{aligned} \quad (15)$$

Notice  $p(\theta|s, \lambda, \lambda')$  still follows same distribution as  $H$  with the same sufficient statistics  $(\eta(\mu), (-a(\mu)))$  and updated natural parameters

$$\lambda \leftarrow \lambda + t(s), \quad \lambda' \leftarrow \lambda' + \gamma. \quad (16)$$

In our case, we have  $\mu \sim H = N(\mu_0, \sigma_0^2)$  and  $s \sim N(\mu, \sigma^2)$ . Hence, we observe that the variational posterior distribution  $\nu_k$  in equation (10) is still Gaussian distributed since the base distribution  $H$  is a conjugate prior of  $p(s_i|\boldsymbol{\mu})$ . Consequently, we have  $t(s) = s/\sigma^2$ ,  $\gamma = 1/\sigma^2$ ,  $\lambda = \mu_0/\sigma_0^2$ , and  $\lambda' = 1/\sigma_0^2$ .

#### 4.4 Incremental Learning of Dirichlet Process Gaussian Mixture Model

One of the major challenges of online social network is that the data is highly dynamic with different update operations such as insertion of a new tie or a change of an existing tie that may occur in an extremely high frequency. A traditional algorithm will require rerun on

not only the changed part of data but also the unchanged part as well. This is a waste of time and resources and is especially problematic if data are enormous in size. A truly incremental algorithm should not only be able to deal with both inserting a new social tie and deleting an old social tie, but also will only require adding computational cost for the changed data. Overall, any change of data can typically be classified as one of these three actions: (1) inserting a new social tie, which involves adding a new similarity measure of the social tie; (2) deleting a social tie, which removes an existing similarity measure of the social tie; and (3) changing an existing social tie, which involves both removing the old similarity measure and adding an updated similarity measure. Furthermore, any change in a single social tie also has a ripple effect on neighbor ties. Therefore, all the adjacent ties will require the action of changing an existing tie as well. We extend Lin’s online algorithm [33], which only involve adding information, to a fully incremental algorithm by allowing both adding and removing information.

#### 4.4.1 Insertion Algorithm

To initialize the Dirichlet process, assume we observe the first social tie strength  $s_1$ . Since there is no existing cluster at this point, the variational distribution  $\rho(z_1 = 1) = 1$ , which is a variational distribution to represent  $p(z_1 = 1|s_1)$ . We also have  $s_1 \in C_1$  and  $|C_1| = 1$ . Recall that both our base distribution  $H$  and the  $p(s_1|z_1, \mu_1)$  are Gaussian distributions; that is,  $H$  is the conjugate prior for  $p(s_1|z_1, \mu_1)$ . Based on what we have established in 4.3.2, we can

update the variational posterior distribution  $\nu_1^{(1)}$  as follows:

$$\begin{aligned}
\nu_1^{(1)} &\propto H(d\mu) \prod_{i \in C_1} p(s_i | \boldsymbol{\mu}) \\
&\propto p(\mu_1 | \mu_0, \sigma_0^2) p(s_1 | z_1 = 1, \mu_1) \\
&\propto \exp\left(\frac{\mu_0}{\sigma_0^2} \mu_1 + \frac{1}{\sigma_0^2} \mu_1^2\right) \exp\left(\frac{s_1}{\sigma^2} \mu_1 + \frac{1}{\sigma^2} \mu_1^2\right) \\
&\propto \exp\left(\left(\frac{\mu_0}{\sigma_0^2} + \frac{s_1}{\sigma^2}\right) \mu_1 + \left(\frac{1}{\sigma_0^2} + \frac{1}{\sigma^2}\right) \mu_1^2\right) \\
&\doteq p(\mu_1 | \lambda_1^{(1)}, \lambda_1^{\prime(1)}).
\end{aligned}$$

Here, we define  $\nu_k^{(i)}$  as the variational posterior distribution of point mass of  $\mu_k$  in component  $k$  at  $i$ -th iteration,  $(\lambda_k^{(i)}, \lambda_k^{\prime(i)})$  are the natural parameters for  $\nu_k^{(i)}$ , and  $\rho_k^{(i)} = \rho(z_i = k)$ . Due to the convenience of calculation, we keep track of the natural parameters  $(\lambda_k^{(i)}, \lambda_k^{\prime(i)})$  instead of the mean and the variance of  $\nu_k^{(i)}$ . However, it should be straightforward to see that  $\nu_k^{(i)}$  is still Gaussian distributed and  $\nu_k^{(i)} = N\left(\frac{\lambda_k^{(i)}}{\lambda_k^{\prime(i)}}, \frac{1}{\lambda_k^{\prime(i)}}\right)$ .

After seeing first observation  $s_1$ , the rest of data are introduced sequentially. At the same time,  $\rho_k^{(i)}$ ,  $C_k$ , and  $\nu_k^{(i)}$  are updated accordingly at each iteration for every  $k$ . Suppose we observe a total of  $N$  observations from social ties, and obtain  $K$  clusters with updated parameters  $(C_1, \dots, C_K)$ ,  $(\nu_1^{(N)}, \dots, \nu_K^{(N)})$ , and  $(\rho^{(1)}, \dots, \rho^{(N)})$  (where  $\rho^{(i)} = \{\rho_1^{(i)}, \dots, \rho_K^{(i)}\}$ ). To interpret a new observation,  $s_{N+1}$  can either belong to one of the existing  $K$  components (i.e.,  $z_{N+1} = k$ ,  $k \in \{1, \dots, K\}$ ) or to a new component (i.e.,  $z_{N+1} = K + 1$ ) with newly introduced  $\rho_k^{(N+1)}$ , and  $\nu_k^{(N+1)}$ . Similar to (9), we then have the iterative posterior distribution  $p(z_{N+1}, \mu_{1:K+1} | s_{1:N+1})$  following:

$$p(z_{N+1}, \mu_{1:K+1} | s_{1:N+1}) \tag{17}$$

$$\propto p(z_{N+1}, \mu_{N+1} | s_{1:N}) p(s_{N+1} | z_{N+1}, \mu_{N+1}) \tag{18}$$

$$\approx q(z_{N+1} | \rho^{(1)}, \dots, \rho^{(N)}, \nu_1^{(N)}, \dots, \nu_K^{(N)}) p(s_{N+1} | z_{N+1}, \mu_{1:K+1}) \tag{19}$$

$$\doteq q(z_{N+1}, \mu_{1:K+1} | \rho^{(1)}, \dots, \rho^{(N+1)}, \nu_1^{(N+1)}, \dots, \nu_K^{(N+1)}) \tag{20}$$



Here,  $\doteq$  denotes “is defined as”. Then, to minimize the Kullback-Leibler divergence between (18) and (19), we can calculate  $\rho_k^{(N+1)}$  and  $\nu_k^{(N+1)}$  by [33]:

$$\rho_k^{(N+1)} \propto \begin{cases} |C_k| \int_{\mu_k} p(s_{N+1}|\mu_k) \nu_k^{(N)}(d\mu_k) & (k \leq K), \\ \alpha \int_{\mu_k} p(s_{N+1}|\mu_k) H(d\mu_k) & (k = K+1). \end{cases} \quad (21)$$

$$\nu_k^{(N+1)} \propto \begin{cases} \hat{H}(d\mu_k) \prod_{i=1}^{N+1} p(s_i|\mu_k) \rho_k^{(i)} & (k \leq K), \\ \hat{H}(d\mu_k) p(s_{N+1}|\mu_k) \rho_k^{(N+1)} & (k = K+1). \end{cases} \quad (22)$$

In particular, we can rewrite the second part of (21) by:

$$\begin{aligned} & \int_{\mu_k} p(s_{N+1}|\mu_k) \nu_k^{(N)}(d\mu_k) \\ &= \int_{\mu_k} p(s_{N+1}|\mu_k) p(\mu_k|\lambda_k^{(N)}, \lambda_k^{\prime(N)}) d\mu_k \\ &= \int_{\mu_k} \exp\{(\lambda_k^{(N)} + t(s_{N+1}))\eta(\mu_k) + (\lambda_k^{\prime(i)} + \gamma)(-a(\mu_k)) \\ &\quad - A(\lambda_k^{(i)}, \lambda_k^{\prime(i)} + B(\mu_k) + b(s_{N+1}))\} d\mu_k \\ &= \exp\{A(\lambda_k^{(N)} + t(s_{N+1}), \lambda_k^{\prime(N)} + \gamma) - A(\lambda_k^{(N)}, \lambda_k^{\prime(N)} + b(s_{N+1}))\} \\ &\doteq \phi(\lambda_k^{(N)}, \lambda_k^{\prime(N)}) \end{aligned} \quad (23)$$

Hence, we can calculate the  $\phi(\lambda_k^{(i)}, \lambda_k^{\prime(i)})$  part in (21) by:

$$\exp\{A(\lambda_k^{(n)} + t(s_{n+1}), \lambda_k^{\prime(n)} + \gamma) - A(\lambda_k^{(n)}, \lambda_k^{\prime(n)} + b(s_{n+1}))\} \quad (24)$$

Regarding  $\nu_k^{(N+1)}$  in (22), we can instead derive the update formulas of  $(\lambda_k^{(N+1)}, \lambda_k^{\prime(N+1)})$ , as we previously defined in (16). In particular, once there is more than one cluster introduced, the update formulas for  $(\lambda_k^{(N+1)}, \lambda_k^{\prime(N+1)})$  needed to be adjusted by the likelihood of assigning  $s$  to clusters; that is, adjusted by multiplying  $\rho_k^{(N+1)} = \rho(z_i = k)$ . We have shown in (11) that  $\hat{H} \propto \frac{\sum_{i=1}^n \rho_k^{(N+1)}}{\alpha + N} \nu_k$ . Therefore, for our incremental Dirichlet process mixture model, we

have the following iterative formula:

$$\rho_k^{(N+1)} \propto \begin{cases} |C_k| \phi(\lambda_k^{(N)}, \lambda_k^{\prime(N)}) & (k \leq K), \\ \alpha \phi(\lambda_0, \lambda_0') & (k = K+1). \end{cases} \quad (25)$$

$$\lambda_k^{(N+1)} \leftarrow \lambda_k^{(N)} + \rho_k^{(N+1)} \frac{S_{N+1}}{\sigma^2}, \quad (26)$$

$$\lambda_k^{\prime(N+1)} \leftarrow \lambda_k^{\prime(N)} + \rho_k^{(N+1)} \frac{1}{\sigma^2} \quad (27)$$

The pseudo code of the incremental algorithm for the insertion of any  $n$  tie strengths is described in Algorithm 1.  $\epsilon$  is set as a cut-off value and we only increase the number of clusters from  $K$  to  $K + 1$  if  $\rho_{K+1}^{(i)} > \epsilon$ . As stated in [33], setting  $\epsilon$  is an efficient way to control the number of clusters while still providing freedom for the model to determine the number of clusters.

#### 4.4.2 Deletion Algorithm

Due the fact that base distribution  $H$  is the conjugate prior and Gaussian is a member of the exponential family, calculating  $C_k$  and  $(\lambda_k^{(N+1)}, \lambda_k^{\prime(N+1)})$  are fairly easy and repetitive tasks. In fact, if we keep records of all the  $\rho_k^{(i)}$ , for  $i = 1, \dots, N$ , it is mathematically possible to erase the contribution from a specific social tie to the model. Here, we devise an incremental algorithm for the deletion of any  $m$  social ties in Algorithm 2. Notice the main differences between the insertion and the deletion algorithm are the updates for  $(\lambda_k^{(N+1)}, \lambda_k^{\prime(N+1)})$  and  $|C_k|$ . That is, the plus signs in insertion are replaced by the minus signs in deletion. Due to the iterative nature of the original insertion algorithm, we can obtain the new results as if those social ties have never been read.

---

**Algorithm 1** The insertion algorithm of iDPGMM

---

- 1: Set  $\sigma^2$ ,  $\mu_0$ ,  $\sigma_0^2$ ,  $\alpha$ ,  $\epsilon$ .
  - 2: Initialize  $K = 1$ ,  $p(z_1 = 1|s_1) = 1$ ,  $\lambda_1^{(1)} = (\mu_0/\sigma_0^2 + s_1/\sigma^2)$ , and  $\lambda_1^{\prime(1)} = (\sigma_0^{-2} + \sigma^{-2})$ . ▷  
Only needed in the first run.
  - 3: **for**  $i = 1$  to  $n$  **do**
  - 4:     Update  $\rho_k^{(i)}$  according to Equation (25).
  - 5:     Normalize  $\rho_k^{(i)}$ , for  $k=1, \dots, K+1$ .
  - 6:     **if**  $\rho_{K+1}^{(i)} > \epsilon$  **then**
  - 7:          $|C_k| = |C_k| + \rho_k^{(i)}$ , for  $k = 1, \dots, K$ .
  - 8:          $|C_{K+1}| = \rho_{K+1}^{(i)}$ .
  - 9:         Update  $\lambda_k^{(i)}$  according to Equation (26).
  - 10:         Update  $\lambda_k^{\prime(i)}$  according to Equation (27).
  - 11:          $K = K + 1$ .
  - 12:     **else**
  - 13:         Remove  $\rho_{K+1}^{(i)}$ .
  - 14:         Renormalize  $\rho_k^{(i)}$ .
  - 15:          $|C_k| = |C_k| + \rho_k^{(i)}$ , for  $k = 1, \dots, K$ .
  - 16:         Update  $\lambda_k^{(i)}$  according to Equation (26).
  - 17:         Update  $\lambda_k^{\prime(i)}$  according to Equation (27).
  - 18:     **end if**
  - 19:     Save  $\rho_k^{(i)}$  for future use.
  - 20: **end for**
-

## 4.5 Cosine Similarity

Let us consider simple, undirected graph  $\langle V, E \rangle$ , where  $V$  is a set of vertices; and  $E$  is set of unordered pairs of distinct vertices. For  $e \in E$ , we denote the unordered pair of vertices by  $(v, w)$  for  $e = (v, w) \in E$ , which is called an edge. The neighborhood of a vertex includes all the vertices connected to it by edges. The social network can then be defined upon this graph. We calculate the social tie strengths  $s$  of social network based on the structural similarities of a graph. Cosine similarity is a structural similarity based on the counting of "common neighbors" [39]. For a vertex  $v \in V$ , we denoted its neighbors by  $\gamma(v)$ , the number of neighbors is called the degree of vertex  $v$ , denoted by  $|\gamma(v)|$ . Cosine similarity between two vertices  $v$  and  $w$  is defined as the number of common neighbors normalized by the geometric mean of the degrees of  $v$  and  $w$ ; that is:

$$\sigma(v, w) = \frac{|\Gamma(v) \cap \Gamma(w)|}{\sqrt{|\Gamma(v)||\Gamma(w)|}} \quad (28)$$

The value of cosine similarity ranges from 0 to 1.

As we mentioned in 4.4, any change in a single social tie also has a ripple effect on neighbor ties. For example, consider a graph demonstrated by the top of Figure 6. Suppose we add an additional edge  $(G, F)$  to to graph. Because cosine similarities solely utilize  $\gamma(G)$  and  $\gamma(F)$ , we only need to re-calculate cosine similarities of edges  $(G, F)$ ,  $(B, G)$ ,  $(E, G)$ ,  $(H, F)$  and  $(E, F)$ , as illustrated by different color of edges by the right side of Figure 6. Our incremental version of DPGMM are then able to run additional iterations for 5 edges instead

---

**Algorithm 2** The deletion algorithm of iDPGMM

---

- 1: **for**  $i = 1$  to  $m$  **do**
  - 2:   Recall  $\rho_k^{(i)}$  for those  $m$  social ties.
  - 3:    $|C_k| = |C_k| - p(z_i = k|s_i)$ , for  $k = 1, \dots, K$ .
  - 4:    $\lambda_k^{(i)} \leftarrow \lambda_k^{(i)} - p(z_i = k|s_i)t(s_i)$ .
  - 5:    $\lambda_k^{l(i)} \leftarrow \lambda_k^{l(i)} - p(z_i = k|s_i)\gamma$ .
  - 6: **end for**
-

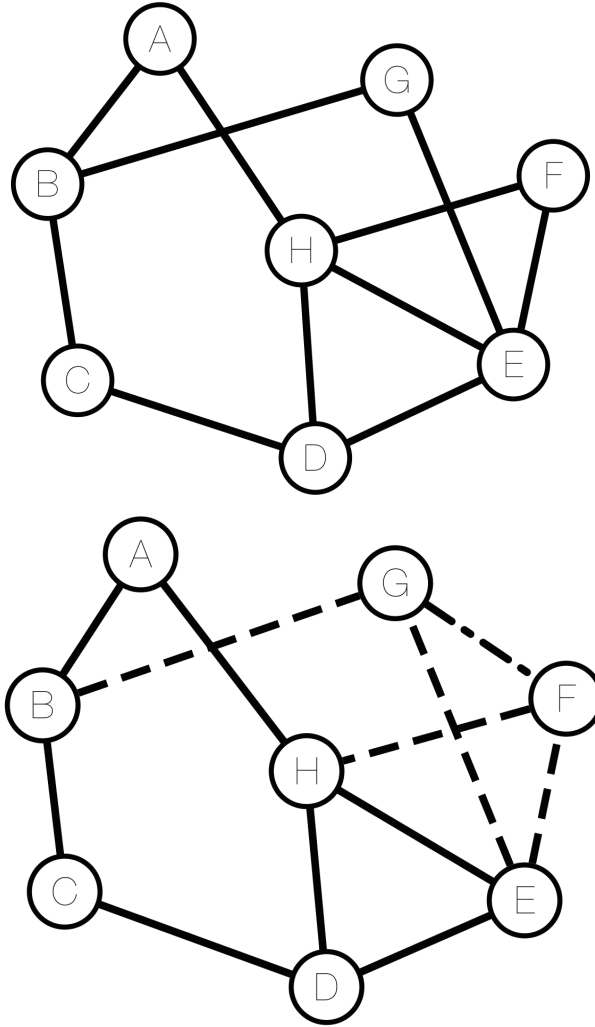


Figure 6: The demonstration on incremental learning

of the whole 12 edges. The time and resources saved will be dramatic if the change of network is relatively small in a big social network.

#### 4.6 Complexity Analysis

In this chapter, An investigation on the computation complexity is presented by walking through the workflow of our algorithm. Suppose we were given a network with  $N$  edges and  $M$  vertices. Initially, a calculation of cosine similarities over all edges is required. This procedure, according to [58], has an  $O(N)$  running time. Using the cosine similarities as input, we then conduct the proposed incremental Dirichlet process Gaussian mixture model

(iDPGMM) algorithm to evaluate the tie strength. During the first run, only insertion algorithm is preformed. Each insertion needs to access  $K$  current components, this has to be operated over all edges. Thus, the total running time at this phase is  $O(NK)$ .

After first full run, for each edge addition or deletion action in social network, as stated in chapter 4.5,  $O(|\gamma(v)| + |\gamma(w)| - 1)$  edges needs to be updated on cosine similarity. However, according to [4], most of the real networks—including World Wide Web, citation network, social network, word co-occurrence network, co-authorship network, etc.—are scale-free networks, meaning that there only a few (such as 3 or 4) neighbors for most vertices. After the calculation of cosine similarities for certain edges, the process of tie strength evaluation then involves both insertion and deletion algorithm as atomic operations. First, we need to delete outdated cosine similarities from the record, then insert the updated ones back. The running time is hence  $O(2K(\gamma(v) + \gamma(w) - 1))$ . Therefore, the complexity of our algorithm is in general  $O(KN)$  and with  $K$  small, can be considered linear in  $N$ .

## 5 Discovering Multiple Social Ties for Characterization of Individuals in Online Social Networks

### 5.1 Datasets

We will use some publicly available social network data about community structures for our experiment. Table 9 shows the basic statistics of the data sets. In particular, only NCAA football data has multiple types of ties ground truth. We now describe them in detail as following.

NCAA Football [58]: The National Collegiate Athletic Association (NCAA) divides 115 schools into eleven conferences. Here, a tie is formed when two school play against each other. In addition, there are four independent schools at this top level: Army, Navy, Temple, and Notre Dame; they are hubs. Each Bowl Subdivision School plays against schools within their own conference (intra-conference ties), against schools in other conferences (inter-conference

Table 9: Datasets statistics

Dataset	Number of Vertices	Number of Edges	Ground Truth
NCAA Football	180	787	Yes
Bible	79	290	No
DBLP Ego	51	130	No
Retweet	48,106	56,334	No
Higgs	456,631	12,508,442	No

ties), and against lower division schools or independent schools (special ties). The network contains 180 vertices (119 Bowl Subdivision schools and 61 lower division schools) interconnected by 787 edges.

**Bible:** We create the network of coappearances of characters in the same chapter of Bible. We prune characters who coappeared less than 3 times to concentrate on more significant connections

**Retweet:** We create this dataset by starting with a set of reporters from 12 news agencies—ABC, The Associated Press, BBC, Bloomberg, CNN, Financial Times, The Guardian, NPR, The New York Times, Reuters, The Washington Post, and The Wall Street Journal. Then, we go through each retweet message in the month of June, 2015. For a given retweet message in which person  $A$  retweeted person  $B$ , if at least one of them is from the starting reporter set, a edge  $(A, B)$  was defined and its frequency was incremented by 1. While the edges are originally directed, we treat them as undirected edges. We might investigate a directed network in the future. At the end, we have the information about number of retweets between two persons and organizations each person belonged to. If a person did not belong to one of the 12 news agencies, the organization of person is labeled as “other”.

**Higgs [15]:** This Twitter friends/followers social network is constructed after monitoring the spreading processes on Twitter before, during and after the announcement of the discovery of a new particle with the features of the elusive Higgs boson on 4th July 2012. The messages posted in Twitter about this discovery between 1st and 7th July 2012 are consid-

ered. The network contains friends/followers social relationships among all users involved in the above activities. Again, While the edges are originally directed, we treat them as undirected edges. Notice that we purposefully choose Higgs dataset solely for testing the speed of each algorithm due to its size.

## 5.2 Reference algorithms

We compare our iDPGMM algorithm with following algorithms in terms of the accuracy and running time:

- VBEM: It is a variational inference algorithm for finite Gaussian mixture model (Figure 5(a)) based on chapter 10.2 of Bishops' book [9]. Matlab codes are written by Emtiyaz Khan, June 2007, and can be downloaded here (<http://www.cs.ubc.ca/~murphyk/Software/VBEMGMM/index.html>). The main differences between VBEM and iDPGMM are (1) VBEM requires fixed the number of clusters; (2) VBEM is not an incremental algorithm.
- Greedy: It characterizes social ties into strong or weak ties based on the Strong Triadic Closure (STC) principle and works by constructing a vertex cover of the graph in a greedy fashion [49]. A online version of Greedy is called MultiGreedy in which Greedy algorithm is repeatedly used on the leftover weak ties after each run. Therefore, a predetermined number of runs will decide the number of cluster at the end. The codes are written in Java and we obtain the codes directly from the original authors [49].

## 5.3 Cluster Assignments

In order to compare the classification accuracy, we need to develop a framework to assign tie into components. The Greedy algorithm has built-in classification ability. For DPGMM and VBEM, we simply compare the log-likelihood of each social ties to all components. That is, we calculate the Gaussian probability density function with the final version of  $\mu_k$  (i.e., we



calculate  $P(s_i|\mu_k, \sigma^2)$ , for  $i = 1, \dots, N$ , and  $k = 1, \dots, K$ ). We then assign each social tie to the cluster with the highest  $P(s_i|\mu_k, \sigma^2)$ . For instance, if  $P(s_{10}|\mu_2, \sigma^2)$  has the highest values among all the  $P(s_{10}|\mu_k, \sigma^2)$ , then tie strength  $s_{10}$  is assigned to cluster 2.

#### 5.4 Evaluation criteria

The detected types of social ties will be evaluated in terms of accuracy and efficiency. The accuracy of the types of social ties will be measured in terms of community structures following the same evaluation method as proposed in [49]. In the following we describe the measures that can be used for the dataset where a ground truth about the type of social ties in terms of community structures is given. For instance, the ground truth of the types of social ties has three categories including intra-community ties, inter-community ties, and special ties to individuals playing special roles such as hubs—which are denoted by  $T_{intra}$ ,  $T_{inter}$ , and  $T_{special}$  respectively, and let  $E_{intra}$ ,  $E_{inter}$ , and  $E_{special}$  denote the corresponding set of edges obtained by the proposed algorithm. Then, we can define the precision  $P_{type}$  and recall  $R_{type}$  for each type of social ties as follows:

$$P_{type} = \frac{|T_{type} \cap E_{type}|}{E_{type}} \quad \text{and} \quad R_{type} = \frac{|T_{type} \cap E_{type}|}{T_{type}}$$

where  $type = \{intra, inter, special\}$ . In addition, an F measure is calculated to compare the overall performance our algorithm with others.

$$F_{type} = 2 \cdot \frac{P_{type} \cdot R_{type}}{P_{type} + R_{type}}$$

Furthermore, the result of the proposed model is a partition of social ties denoted by  $\mathbf{C} = \{C_1, C_2, \dots, C_K\}$ , which can be compared with the ground truth partition of the social ties in the dataset denoted by  $\mathbf{T} = \{T_1, T_2, \dots, T_K\}$ . One common way to measure cluster quality is to compute the mutual information between  $\mathbf{C}$  and  $\mathbf{T}$ . To this end, let  $P_{CT}(i, j) = \frac{|C_i \cap T_j|}{n}$  be the probability that a randomly chosen object belongs to cluster  $C_i$

in  $\mathbf{C}$  and  $T_j$  in  $\mathbf{T}$ . Also, let  $P_C(i) = \frac{|C_i|}{N}$  be the probability that a randomly chosen object belongs to cluster  $c_i$  in  $\mathbf{C}$ ; define  $P_T(j) = \frac{|T_j|}{N}$  similarly. Then we have

$$I(\mathbf{C}, \mathbf{T}) = \sum_{i=1}^K \sum_{j=1}^K P_{CT}(i, j) \log \frac{P_{CT}(i, j)}{P_C(i)P_T(j)}$$

The value of mutual information is between 0 and minimum of the entropies. Unfortunately the maximum of mutual information can be achieved by using many small clusters. A remedy of this problem is to use the normalized mutual information (NMI),

$$NMI(\mathbf{C}, \mathbf{T}) = \frac{I(\mathbf{C}, \mathbf{T})}{(H(\mathbf{C}) + H(\mathbf{T}))/2},$$

where  $H(\mathbf{C})$  and  $H(\mathbf{T})$  are entropies. NMI lies between 0 and 1.

Another accuracy measure for data clustering is adjusted Rand index (ARI), which is the version of Rand index corrected for chance. Let  $n_{ij} = |C_i \cap T_j|$ ,  $a_i = |C_i|$ , and  $b_j = |T_j|$ .

$$\begin{aligned} ARI(C, T) &= \frac{Index - ExpectedIndex}{MaximumIndex - ExpectedIndex} \\ &= \frac{\sum_{i,j} \binom{n_{ij}}{2} - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{a_i}{2} + \sum_j \binom{b_j}{2}] - [\sum_i \binom{a_i}{2} \sum_j \binom{b_j}{2}]/\binom{n}{2}} \end{aligned}$$

Unlike a typical Rand Index, which lies between 0 and 1, ARI can also yield negative value if index is smaller than Expected index, which causes the numerator to be negative.

## 5.5 Results

The performance of the proposed model is evaluated in terms of the accuracy and the efficiency by using both benchmark data and real online social network data. The goal of performance evaluation is to make sure that the result achieved by using the proposed approach matches with the ground truth about social ties in terms of community structures. We run different experiments to demonstrate this. All the experiments are conducted on

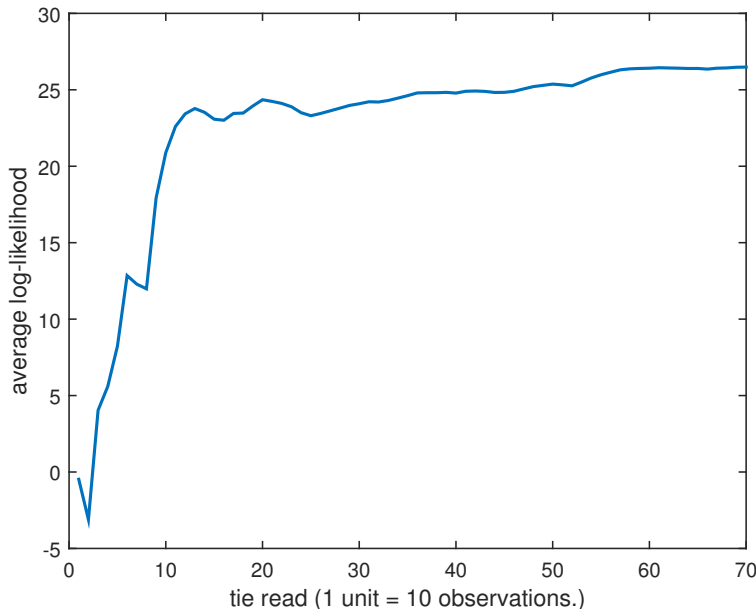


Figure 7: The average log-likelihood from 10-fold cross validation on NCAA Football

a HP DL980 server with 8 Intel(R) Xeon(R) CPU E7- 4870 @ 2.40GHz (each CPU has 10 cores) and 4 TB memory.

### 5.5.1 Model Convergence

We typically set  $\sigma^2 = 0.1$ ,  $\mu_0 = 0$ ,  $\sigma_0^2 = 0.5$ ,  $\alpha = 0.5$ , and  $\epsilon = 0.1$ <sup>1</sup>. We monitored the convergence of the model using average log-likelihood on held-out data from 10-fold cross validation. The likelihood of held-out data can be calculated by:

$$\begin{aligned}
 p(\mathbf{s}_{test}|\mathbf{s}_{train}) &= \sum_k p(\mathbf{z}|\mathbf{s}_{train})p(\mathbf{s}_{test}|\hat{\lambda}_k, \hat{\lambda}'_k) \\
 &\propto \sum_k \frac{|C_k|}{n} p(\mathbf{s}_{test}|\hat{\lambda}_k, \hat{\lambda}'_k)
 \end{aligned}$$

where  $(\hat{\lambda}_k, \hat{\lambda}'_k)$  is natural parameters after running all training data and  $p(\mathbf{s}_{test}|\hat{\lambda}_k, \hat{\lambda}'_k)$  has a probability density function of  $N(\frac{\lambda_k^{(i)}}{\lambda_k^{(i)}}, \frac{1}{\lambda_k^{(i)}})$ . As we can see in Figure 7 from cross validation

<sup>1</sup>For all experiments except Retweet, we use  $\sigma^2 = 0.1$ ,  $\mu_0 = 0$ ,  $\sigma_0^2 = 0.5$ ,  $\alpha = 0.5$ , and  $\epsilon = 0.1$ . For Retweet data, we only adjust  $\epsilon$  to 0.15 in order to reduce redundant components.

on NCAA Football, the model is relatively stable after seeing around 120 data. We see similar early convergences in all our other experiments.

### 5.5.2 Classification Accuracy

As we mentioned in 5.3, we develop a framework to compare classification accuracy among different algorithms. Hence, we use the only dataset with ground truth of multiple types of ties to test our results. Fortunately, NCAA football dataset is among the very few that contains the ground truth of multiple types of social ties—intra-conference ties, inter-conference ties, and special ties. Under our model specification, iDPGMM is able to obtain multiple types of ties with  $(\mu_1, \mu_2, \mu_3) = (0.22, 0.27, 0.57)$ . Because the values of the cosine similarities are associated with the strength of the tie, we instinctively associate  $\mu_3$  with intra-conference ties,  $\mu_2$  with special ties, and  $\mu_1$  with inter-conference ties. For MultiGreedy algorithm, we let it run two times in order to generate 3 types of ties. Similar to [49], the strong ties from the first run is associated with intra-conference ties. The Greedy algorithm is then reused again on the weak ties of the first runs. The strong ties from the second run is associated with special tie, and the weak ties from the second run is associated with inter-conference ties. For VBEM, like MultiGreedy, it requires a predetermined number of clusters. After we set the number of clusters equal to 3, we obtain three Gaussian models with various degree of  $\mu_k$ . Again, we associate the largest  $\mu_k$  to intra-conference ties, middle  $\mu_k$  to special ties, and weakest  $\mu_k$  to inter-conference ties. We then calculate several evaluation criteria to compare the outcomes—precision, recall, f-measure, normalized mutual information, and adjusted Rand index. The results as shown in Table 10, Table 11, Table 12, and Table 13.

Similar to what Sintos et al. has shown in [49], the MultiGreedy algorithm generally produces impressive precision on strong ties (intra-conference ties) and recall on weak ties (inter-conference ties). However, other measures from MultiGreedy are often less accurate. For VBEM, while there have been one instance that VBEM outperforms iDPGMM (i.e.,

Table 10: Number of ties found in NCAA Football dataset

	Ground Truth	iDPGMM	MultiGreedy	VBEM
Inter-conf.	207	182	369	183
Special	123	149	98	25
Intra-conf.	457	456	320	579

Table 11: Precision and Recall on NCAA Football dataset

	iDPGMM		MultiGreedy		VBEM	
	Precision	Recall	Precision	Recall	Precision	Recall
Inter-conf.	<b>0.87</b>	0.77	0.46	<b>0.82</b>	<b>0.87</b>	0.77
Special	<b>0.63</b>	<b>0.76</b>	0.39	0.31	0.36	0.07
Intra-conf.	<b>0.99</b>	0.98	<b>0.99</b>	0.69	0.79	<b>1</b>

Table 12: F-measures on NCAA Football dataset

	iDPGMM	MultiGreedy	VBEM
$F_{Inter}$	<b>0.82</b>	0.59	<b>0.82</b>
$F_{Special}$	<b>0.69</b>	0.34	0.12
$F_{Intra}$	<b>0.99</b>	0.82	0.88

Table 13: AIR and NMI on NCAA Football dataset

	iDPGMM	MultiGreedy	VBEM
ARI	<b>0.83</b>	0.31	0.54
NMI	<b>0.70</b>	0.32	0.48

Recall of intra-conference ties), the differences are insignificant. Furthermore, VBEM is the worst to identify special ties—it only identified 25 out of 123 true special ties. For iDPGMM, it has the best score in  $F_{Special}$ ,  $F_{Intra}$ , ARI, and NMI, suggesting it is the best in overall performance for this example. In addition, iDPGMM is the best in finding special ties. Recall that DPGMM is an unsupervised algorithm without the need to specify number of clusters. Under our model specification, we are still able to discover all three types of tie for NCAA Football with good accuracy.

### 5.5.3 Deletion of Social Ties

As we have shown, changing social ties may only lead to a small number of changes for the cosine similarities. To demonstrate this, we randomly remove one social tie, the edge between vertex 78 and vertex 107 ( $e_{delete}$ ), in NCAA football data. After recalculating cosine similarity based on the new social structure, this removal of a single tie causes a total of 22 changes of cosine similarities from the adjacent ties. Here, we compare the results from the following two methods:

1. Rerun our iDPGMM on the new set of cosine similarities.
2. Utilize our insertion and deletion algorithms to replaced only the changed cosine similarities.

In our second approach, we first utilize our deletion algorithm to remove a total of 23 cosine similarities, including  $e_{delete}$  and 22 others which were affected by  $e_{delete}$ . Then, we utilize our insertion algorithm to add 22 updated cosine similarities back. We record both running time. The first method required 0.253597 seconds while second only 0.021764 seconds in our machine. Not only is the incremental method 11 times faster, we have obtained identical cluster assignments from both methods. This supports the claim that we can save time and computation resources using our method. This time savings will even become much greater if data are truly dynamic and enormous in size, e.g. Facebook, Twitter, etc.

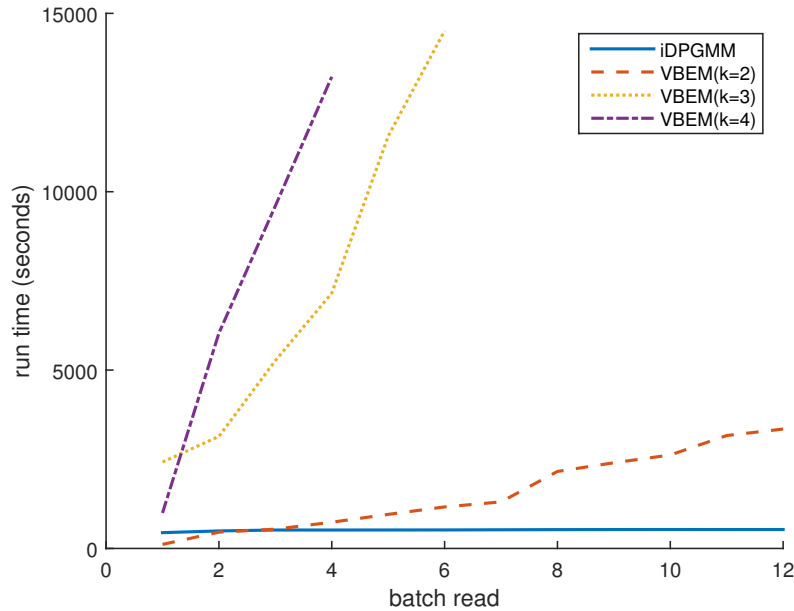


Figure 8: The running times on Twitter

#### 5.5.4 Running time on Higgs data

Traditional models required rerun on all data whenever the data are added or deleted. Therefore, the increase of data will mean the increase of running time. On the other hand, iDPGMM is very suitable for big data because it only needs to run on the changed part of the data while keeping the unchanged part in tact. Consequently, the running time stay relatively stable. We compare our model running time with VBEM on Higgs data, which contain a whopping 12,508,442 ties. Notice iDPGMM and VBEM are the only two model-based algorithms among all three. Comparing to the Greedy algorithm is not fair since the two are built on different hypotheses. Because VBEM is a truncated algorithm, it is required to set a initial number of clusters and the initial number should be greater than the expected number of clusters. As long as the initial number is large enough, the results from different setting should be the same. However, the running time grows as the number of initial cluster increases. We first separate the Higgs data into 12 equal size batches. We run our iDPGMM and VBEM with different initial clusters ( $k=2, 3,$  and  $4$ ) while adding one batch of the Higgs data at a time. All algorithms run under the same computer hardware and software (i.e.,

Table 14: Cluster outcomes for Bible network

Cluster	1	2	3	4
$\mu_k$	0.42	0.69	0.88	0.24
Count	105	56	80	49

Matlab) configuration. In Figure 8, as expected, the running time of iDPGMM is stable because iDPGMM only runs on the additional data. On the contrary, the running times of all VBEMs are increasing when a new batch of data is inserted. Note the running time for VBEM with  $k=3$  and 4 is partially plotted in Figure 8 for better comparison.

### 5.5.5 Multiple Types of Social Ties in Bible

Under our approach, iDPGMM characterizes social ties into multiple types with various levels of estimated mean cosine similarities (i.e.,  $\mu_k$ , for  $k = 1, \dots, K$ ). Hence, the strength of each type of social tie in iDPGMM can simply be determined by  $\mu_k$ . In traditional topic modeling on text documents, documents are profiled based on their topic distributions. Similarly, we can profile individuals by their cluster distribution found by iDPGMM. Combining the knowledge of network constraints on the importance of the network, we can profile each person under multiple type assumption and project them based on their profile. Here, our goals can be summarized by:

1. Identify legitimate multiple types of social ties in a large social network.
2. Characterize individuals using the cluster distribution generated from iDPGMM.

In the Bible dataset, a tie is only formed if two persons have more than 3 coappearances in one chapter of the Bible, so that we can concentrate on meaningful connections. At the end, our iDPGMM identifies 4 clusters with various degree of  $\mu_k$  (Table 14).

As seen in Table 15, each type of social tie can be found to associate with certain social traits. The tie leaders in cluster 3, the strongest tie, are all apostles. That is, the strongest relationship classified in cluster 3 is closely related to people who are in a highly connected



Table 15: Tie leaders in each cluster of the bible network

$C_3(0.88)$		$C_2(0.69)$	
Name	Count	Name	Count
John	12	Peter	17
Andrew	11	David	10
James(son of Zebedee)	11	Pilate	7
Matthew	11	John the Baptist	5
Philip(the apostle)	11	Herod(Antipas)	5
$C_1(0.42)$		$C_4(0.24)$	
Name	Count	Name	Count
Jesus	29	Jesus	44
Paul	15	Abel	2
Moses	14	Paul	2
John the Baptist	12	Aaron	1
Abraham	11	Adam	1

Note:  $\mu_k$  of each cluster is shown between parentheses.

community (i.e., Jesus’ apostles). Along with David, Pilate, John the Baptist, and Herod, Peter has the most ties in cluster 2, that is, the second strongest relationship. This type of relationship is associated with leaders in communities. In other words, they are usually related to a highly connected community (e.g., a nation, an organization, or a church) yet they still often need to communicate with other “outsiders”. Indeed, Peter is a church leader, David is a king of a nation, Pilate is a ruler of a land, and John the Baptist is a leader of a religious group. They are all leaders in their own group and often have need to communicate with people who are in different communities. In cluster 1 and cluster 4 (i.e., the weakest relationships), we have the people who are associated with a wide range of communities. In particular, cluster 4, the weakest relationship, is nearly exclusive to Jesus. As we discussed previously, these “social brokers” (i.e., people who many weaker ties) often play a significant role of translating information among different groups and is key to hold the whole network together. Jesus, being the key figure of the whole book, is definitely qualified as a “social broker”.

Each of the top tie leaders—John, Peter, and Jesus—have the most connections regarding their own category, and they also present different type of social status regarding to the

Table 16: Tie leaders cluster distribution

Name	$C_3(0.88)$	$C_2(0.69)$	$C_1(0.42)$	$C_4(0.24)$	Sum
John	12	3	4	0	19
Peter	1	17	7	1	26
Jesus	0	2	29	44	75



Figure 9: John's surrounding network

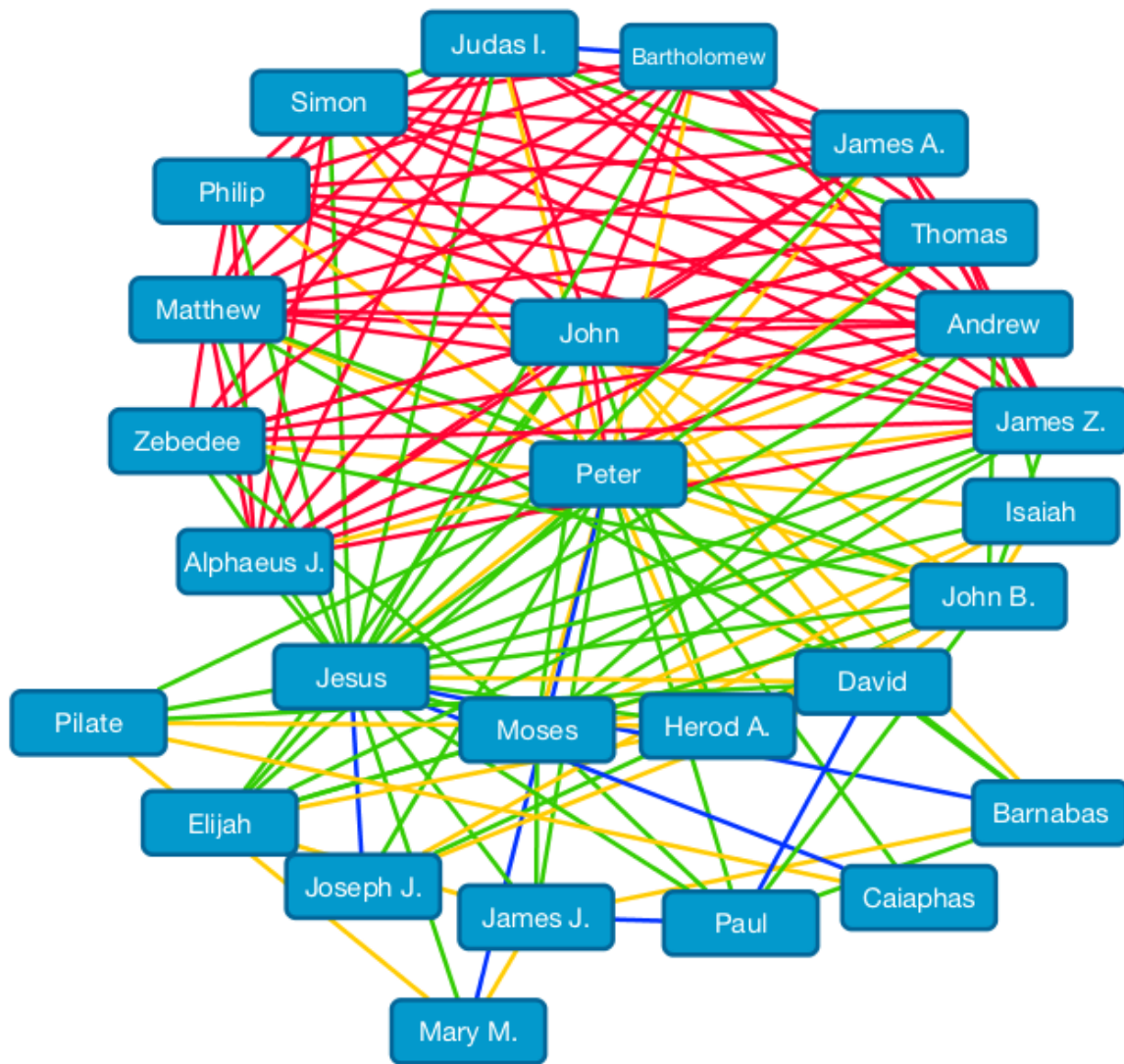


Figure 10: Peter's surrounding network



network. For John, who has significant amounts of ties in cluster 3 (strongest) and no tie in cluster 4 (weakest) (Table 16, he represents a “local leader” type in a network, who often receives and translates information solely within his own group.

To visualize John’s surrounding community, we pick people whom John has ties with and plot them in Figure 9. Here, the distance between the John and other people represents the value of cosine similarities between John and them—the closer the stronger. The color of ties represents different types of ties: red represents the cluster 3 tie (strongest,  $\mu_3 = 0.88$ ), yellow represents the cluster 2 tie (less strong,  $\mu_2 = 0.69$ ), green represents the cluster 1 tie (weaker,  $\mu_1 = 0.42$ ), and blue represents the cluster 4 tie (weakest,  $\mu_4 = 0.24$ ). Figure 9 illustrates that John is indeed in a closed “apostle” community where there exists plenty of red ties (i.e., strongest) surrounding John. We called it “apostle” community since people who have red ties with John are either one of the Jesus’ apostles or one of the apostle’s father.

In Peter’s surrounding network (see Table 16), besides one cluster 3 tie (strongest,  $\mu_3 = 0.88$ ) with John and one cluster 4 tie (weakest,  $\mu_4 = 0.24$ ) with Mary Magdalene, Peter has significant amounts of cluster 2 (less strong,  $\mu_2 = 0.69$ ) and cluster 1 ties (less weak,  $\mu_1 = 0.42$ ). As a result, Peter represents a “regional leader” type in a social network. While still connecting with their own communities, “regional leader” type also interact with people outside—hence, forming less strong relationships. We can see that Figure 10 closely illustrates our assumption of a “regional leader” type. Finally, the dominant number of green (less weak) and blue (weakest) ties make Jesus a “global leader” type—a person who bridges different communities together (Figure 11).

### 5.5.6 Multiple Types of Social Ties in Twitter

In Retweet dataset, retweets are generally used to share information with tweeters’ followers. By tracking the tweeters and retweeters, we have the traces of information access and flow in a online social network. In our run, 3 types of ties are identified for Twitter network

Table 17: iDPGMM outcomes for Twitter network

Cluster	1	2	3
$\mu_k$	0.06	0.59	0.37
Count	50674	779	4881

Table 18: Persons who have most ties in each cluster of Twitter network

$C_2(0.59)$			$C_3(0.37)$			$C_1(0.06)$		
ID	Org.	Count	ID	Org.	Count	ID	Org.	Count
samjordison	guardian	7	<b>rachelapoly</b>	ap	43	<b>mark_beech</b>	bloomberg	23737
<b>jimschachter</b>	nyt	6	bindelj	guardian	36	spiegelpeter	ft	2461
abby_aguirre	nyt	6	dancancel	bloomberg	34	rolandsmartin	cnm	2156
wayneparryac	ap	6	davehill	guardian	32	paulmasonnews	bbc	1597
vranicawsj	wsj	5	oliverburkeman	guardian	31	jaketapper	abc	1246

(Table 17). In the retweet network, we observe that counts of social ties in clusters are in contrast to the strengths of social ties—most of the ties belong to weak relationships while very few ties belong to strong relationships. Following our previous conclusion, strong ties are associated with more local, highly connected communities while weak ties are associated with more global, less connected communities. Here, we focus on several tie leaders (i.e., *mark\_beech*, *jimschachter*, and *rachelapoly*) in each cluster to see if specific retweet patterns can be detected (Table 18).

Similar to how we analyze Bible network, we can identify “local leader” type, “regional leader” type, and “global leader” type in Retweet network. As we can see in Table 19, each example has different distributions of social ties. *mark\_beech*, being a well-known art journalist, has the most cluster 2 ties of all and his tweets are widely retweeted. Hence, *mark\_beech* is a “global leader” type in Retweet network. *rachelapoly* is a correspondent

Table 19: Cluster distributions for selected examples in Twitter network

Name	$C_2(0.59)$	$C_3(0.37)$	$C_1(0.06)$	Sum
jimschachter	6	1	0	7
rachelapoly	0	43	5	48
mark_beech	0	0	23737	23737

Table 20: iDPGMM outcomes for Kleinberg’s ego-network

Cluster	$C_1$	$C_2$	$C_3$	$C_4$
$\mu_k$	0.42	0.31	0.74	0.84
Count	68	86	66	40

for The Associated Press and she cover politics and breaking news in Washington state. Hence, her inference on the network is mostly related to a specific region (i.e., Washington state). With the most cluster 3 ties, *rachelapoly* is indeed our “regional leader” type in the network. Finally, *jimschachter* is a radio station host in New York city and hence his social interactions should be mostly about New York City. Hence, with many cluster 1 ties, *jimschachter* is a “local leader” type. To this end, we observe that the distributions of social ties of individuals reflect not only the network constraints of the surrounding networks, but also the degree of influences individuals have on the network. The types of social ties ones have will determine whether they have a global, regional, or local influence on the network. In general, the more weaker ties a individual has, the more influential a individual is in the network.

### 5.5.7 Multiple Types of Social Ties in Ego-network

Following work of Sintos et al. [49], we create ego-network for Jon M. Kleinberg from DBLP dataset. An ego-network, as name suggests, is the network containing relationships of a single individual and the ties between the individual and his neighbors. We prune the co-authors who have less than 3 publications together in order to focus on more meaningful results. As Sintos et al. demonstrated, multiple social types can be associated with certain social traits. In our experiment, iDPGMM identifies 4 clusters in Kleinberg’s ego-network (Table 20). We observe that cluster 4 (the strongest) is associated with the collaborations within a single institution—Cornell and IBM. Cluster 3 (less strong) is associated with the collaborations related to multiple institutes—IBM, Yahoo, and Google. Cluster 1 (less weak) is associated with Kleinberg’s closest colleges. Finally, cluster 2 (the weakest) is almost

Table 21: Tie leaders in each cluster for Kleinberg’s ego-network

$C_4(0.84)$		$C_3(0.74)$	
Name	Count	Name	Count
Daniel P. Huttenlocher	3	Ravi Kumar	7
Sridhar Rajagopalan	3	Andrew Tomkins	7
Anupam Gupta	2	Prabhakar Raghavan	6
Amit Kumar 0001	2	Jure Leskovec	4
Moses Charikar	2	Sridhar Rajagopalan	3
$C_1(0.42)$		$C_2(0.31)$	
Name	Count	Name	Count
Éva Tardos	12	Jon M. Kleinberg	42
Jon M. Kleinberg	8	Robert D. Kleinberg	2
Jure Leskovec	7	Éva Tardos	1
Ravi Kumar	6	David Liben-Nowell	1
Prabhakar Raghavan	4	Yuval Rabani	1

Table 22: Tie leaders cluster distribution in Kleinberg’s ego-network

Name	$C_4(0.84)$	$C_3(0.74)$	$C_1(0.42)$	$C_2(0.31)$	Sum
Daniel P. Huttenlocher	3	0	1	1	5
Ravi Kumar	1	7	6	0	14
Andrew Tomkins	1	7	3	0	11
Prabhakar Raghavan	0	6	7	0	10
Éva Tardos	0	1	12	1	14
Jon M. Kleinberg	0	0	8	42	50

exclusively associated to Kleinberg himself.

Apply the same rationale we developed previously in Bible network, individuals can be characterize in Kleinberg’s ego-network in the same way. As seen in Table 22, Kumar, Tomkins, and Raghavan, having the most amount of middle strength ties, have all worked for IBM, Yahoo, and Google overtime and are the “regional leaders” of the network—Being “regional” perfectly reflect their experiences in different companies. Being the Dean and Vice Provost of Cornell Tech, Huttenlocher works closely within the Cornell community and is our “local leader” regarding to the network.

Regarding to weak relationships, Kleinberg unsurprisingly has the most weaker ties and is the true “global leader”, a person who bridge the whole network. Besides Kleinberg himself,



Tardos has the most weaker ties and hence should be the second most important “global leader”. In Sintos’ finding, the tie between Kleinberg and Tardos was classified as a weak relationship. However, Tardos is not only one of the most frequent co-authors with Kleinberg, but she is also a colleague of Kleinberg in the same department of Cornell. Their relationship should definitely be classified as strong, as iDPGMM does in our experiment. This shows the advantage of our approach—which successfully captures the importance of Tardos—and the shortcoming from Greedy+—simply maximizing the number of strong ties sometimes lead to a failure of capture the true strength of the relationship between individuals. Furthermore, with iDPGMM, there is no need to specify the number of clusters in advance. iDPGMM decides the number of cluster itself.

## 6 Discussion and Future Work

Our proposed asymmetric author-topic model is useful in the large-scale genomics data set analysis because of their ability to handle large numbers of potentially interrelated variables, and because of their ability to discern statistical relationships between drugs and their inner pathways. In this study, we first give our rationale on why a probabilistic topic model is suitable for genomic profiling expression, such as the Japanese Toxicogenomics Project database. We have demonstrated that our AAT model can be implemented to explore hidden relationships among different features (treatment, drug, and time-dose) and genes through latent biological processes. The straightforward data preprocessing makes the transition of data format manageable and easy to expand. In fact, the same principle of data preprocessing can also be applied to next-generation sequencing (NGS) technology since microarray expression intensity can be simply replaced by read counts in NGS [60]. Since our model enhances the traditional probabilistic topic modeling approach without altering the core assumptions, our framework can be easily adapted for new probabilistic topic model. For example, if we have labels or classes attached to each treatment, we can again enhance supervised topic models [35] with asymmetric priors and apply the model to a database with the same feature-centric

analysis capacity. Because of the popularity of probabilistic topic modeling, there are many existing and well-built software packages ready to be used, including MALLET. Therefore, the implementation of newer probability topic models should also be straightforward in the future. Moreover, other models can also potentially improve some of the limitations our model has. Although changing a continuous value (i.e., fold change values) into a discrete value (i.e., counts) has been done before [18], this process ultimately decreases the precision of the data. Models like the Gaussian mixture model that supports continuous outcome will eliminate the need of altering data. Another limitation of our model is the need to determine the number of latent biological processes in advanced. While the perplexity analysis ensures a relatively proper number of latent processes were chosen initially, finding an optimal number of latent processes is still difficult and costly. Many nonparametric Bayesian models have been developed, including Hierarchical Dirichlet Processes [52], and Hierarchical Pachinko Allocation [37], and the number of latent processes is automatically determined within the algorithm.

One definite advantage of AAT model is the ability to connect the latent biological processes with functional annotation. By connecting our finding with Kyoto Encyclopedia of Genes and Genomes (KEGG) pathways via DAVID [24], we further increase the interpretability of latent biological processes. Therefore, we are able to browse and interact with TGP data through meaningful and interpretable biological pathway (i.e., glutathione metabolism). Regarding the application on glutathione depletion, acetaminophen is a well-known drug that can potentially cause fatal liver injury due to an overdose. Through our approach, we identify that the alteration of glutathione metabolism at even the middle dose (600 mg) of acetaminophen as early as treatment day three. The conclusion of linkages among pathway glutathione metabolism, acetaminophen, and other 5 drugs are found and confirmed in other studies. This demonstrates the possibility of finding existing or new pathway-like annotation through our proposed model, and the ability to cluster drugs with similar mechanisms of action. It is possible to even predict potential pathways for a new drug by estimating the

probability distribution of latent biological processes under this framework. Our model also has the capability to adapt analysis that put focus on different features of data. We show how to identify the dominant factor in dose and time combinations in our second application through generalized linear model. As animal reduction in experiment becomes a global trend, the outcome of similarity of time-dose combination is a viable approach to reducing animals needed for future study. Overall, AAT model has demonstrated potential to be an accessible and flexible approach for finding hidden patterns in large toxicogenomic data.

In our second study, understanding the dynamic nature of social ties between individuals plays an essential role for many applications including community structure detection, real time event detection and viral marketing. Therefore, it remains as a major task for disciplines such as sociology, education, economics, and psychology. In this study we propose an unsupervised approach to the characterization of social ties. We apply the Dirichlet process Gaussian mixture model for grouping tie strengths into clusters that correspond to different type of social ties. To address update in the data we implement an incremental model inference algorithm for dynamic online social networks. The empirical evaluation using some real social networks demonstrates a superior performance in terms of both accuracy and running time in comparison with other algorithms. In addition, our algorithm doesn't require the number of clusters as a parameter, which is very beneficial for very large dynamic online social networks such as Twitter. Furthermore, our approach demonstrate strong performance on real online social networks as well. In Bible data, iDPGMM successfully identified "global leader", "regional leader", and "local leader" based on the characteristics of one's social ties. In Retweet data, we again demonstrate that the degree of impact one has is linked to the distribution of one's social ties. In DBLP, we explore Kleinberg's ego-network and discover various types of social connections. Our model identified a close colleague of Kleinberg as one of the "global leader", while Greedy algorithm labeled it as a weak relationship.

There are several areas which we would like to explore in the future. First, although our approach is effective, it is not fully Bayesian approach; that is, only  $\mu_k$  is treated as random

variables. A fully Bayesian approach will require a  $\sigma_k^2$  to be random variables and base distribution will be a Gaussian-inverse-gamma distribution in order to preserve the conjugate prior property. On the other hand, a fully Bayesian approach will require additional parameters and hence potentially create overfitting problem. Furthermore, the inverse-gamma distribution has two hyperparameters which have range from greater than 0 to  $\infty$  and are hard to initialize correctly. A true fully Bayesian approach, where all four hyperparameters are updated iteratively, may be possible, but it is hard to see overwhelming benefit because iDPGMM already performs fairly well—sometimes even outperforming others. One may consider other distributions, like Beta or Pareto distributions, since they both have range from 0 to 1, which match the range of cosine similarity. We did build models based on Pareto and Beta, yet each has their own problems. Although many social network are highly skewed, some have significant amounts of strong ties—skewed to the left, which does not fit Pareto distribution well. For Beta distribution, while it may be the most obvious choice, lack of conjugate prior support remains a big issue. In 2011 article [34], Ma proposed a clean closed-form solution. We adapted Ma’s approach yet the performance is highly unstable. By simply changing random number generator, we have 4 completely different outcomes. This is mainly due to the wide range of possible values (again, from greater than 0 to  $\infty$ ) and the lack of proper restrictions for both parameters of Beta. Therefore, a flat base distribution—often used in Dirichlet process—create huge differences in each run.

In Lin’s study [33], cluster pruning and merging are also proposed to handle redundant cluster problems. while we didn’t use pruning or merging in our experiments, we have already added pruning to our model. It is specially useful for a distribution like Beta because it is more vulnerable to overfitting. On the other hand, merging requires a pair-wise similarity measures of  $\rho_k^{1:N}$  for all  $k$ , which is problematic when number of ties  $N$  is going very large.

Besides improvement on the model, iDPGMM can also be extended to other areas, like text document. A Dirichlet process Multinomial mixture model(DPMMM) has just been proposed [59], yet there is still no truly incremental version of DPMMM. Overall, we see

much potential in our work and we plan to explore them in the future.

## References

- [1] C. A. Afshari, H. K. Hamadeh, and P. R. Bushel. The evolution of bioinformatics in toxicology: advancing toxicogenomics. *Toxicological Sciences*, page kfq373, 2010.
- [2] R. Agarwal, L. A. MacMillan-Crow, T. M. Rafferty, H. Saba, D. W. Roberts, E. K. Fifer, L. P. James, and J. A. Hinson. Acetaminophen-induced hepatotoxicity in mice occurs with inhibition of activity and nitration of mitochondrial manganese superoxide dismutase. *Journal of Pharmacology and Experimental Therapeutics*, 337(1):110–118, 2011.
- [3] L. Backstrom and J. Kleinberg. Romantic partnerships and the dispersion of social ties: a network analysis of relationship status on facebook. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 831–841. ACM, 2014.
- [4] B. A.-L. BARABÁSI and E. Bonabeau. Scale-free. *Scientific American*, 2003.
- [5] R. Ben-Shachar, Y. Chen, S. Luo, C. Hartman, M. Reed, and H. F. Nijhout. The biochemistry of acetaminophen hepatotoxicity and rescue: a mathematical model. *Theor Biol Med Model*, 9:55, 2012.
- [6] M. A. Beyer and D. Laney. The importance of ‘big data’: a definition. *Stamford, CT: Gartner*, 2012.
- [7] M. Bicego, P. Lovato, A. Ferrarini, and M. Delledonne. Biclustering of expression microarray data with topic models. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 2728–2731. IEEE, 2010.
- [8] M. Bicego, P. Lovato, A. Perina, M. Fasoli, M. Delledonne, M. Pezzotti, A. Polverari, and V. Murino. Investigating topic models’ capabilities in expression microarray data classification. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 9(6):1831–1836, 2012.
- [9] C. M. Bishop et al. *Pattern recognition and machine learning*. springer New York, 2006.
- [10] D. M. Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012.
- [11] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.
- [12] R. S. Burt. Structural holes and good ideas1. *American journal of sociology*, 110(2):349–399, 2004.

- [13] M. Chen, M. Zhang, J. Borlak, and W. Tong. A decade of toxicogenomic research and its contribution to toxicological science. *Toxicological Sciences*, page kfs223, 2012.
- [14] M.-H. Chung, Y. Wang, H. Tang, W. Zou, J. Basinger, X. Xu, and W. Tong. Asymmetric author-topic model for knowledge discovering of big data in toxicogenomics. *Frontiers in pharmacology*, 6, 2015.
- [15] M. De Domenico, A. Lima, P. Mougel, and M. Musolesi. The anatomy of a scientific rumor. *Scientific reports*, 3, 2013.
- [16] J. DeRisi, L. Penland, P. O. Brown, M. L. Bittner, P. S. Meltzer, M. Ray, Y. Chen, Y. A. Su, and J. M. Trent. Use of a cdna microarray to analyse gene expression patterns in human cancer. *Nature genetics*, 14(4):457–460, 1996.
- [17] D. Easley and J. Kleinberg. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.
- [18] P. Flaherty, G. Giaever, J. Kumm, M. I. Jordan, and A. P. Arkin. A latent variable model for chemogenomic profiling. *Bioinformatics*, 21(15):3286–3293, 2005.
- [19] L. Fu and E. Medico. Flame, a novel fuzzy clustering method for the analysis of dna microarray data. *BMC bioinformatics*, 8(1):3, 2007.
- [20] E. Gilbert. Predicting tie strength in a new medium. In *Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work*, pages 1047–1056. ACM, 2012.
- [21] E. Gilbert and K. Karahalios. Predicting tie strength with social media. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 211–220. ACM, 2009.
- [22] M. S. Granovetter. The strength of weak ties. *American journal of sociology*, pages 1360–1380, 1973.
- [23] 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.
- [24] D. W. Huang, B. T. Sherman, and R. A. Lempicki. Systematic and integrative analysis of large gene lists using david bioinformatics resources. *Nature protocols*, 4(1):44–57, 2008.
- [25] Y. Igarashi, N. Nakatsu, T. Yamashita, A. Ono, Y. Ohno, T. Urushidani, and H. Yamada. Open tg-gates: a large-scale toxicogenomics database. *Nucleic acids research*, page gku955, 2014.
- [26] L. F. James, A. Lijoi, and I. Prünster. Posterior analysis for normalized random measures with independent increments. *Scandinavian Journal of Statistics*, 36(1):76–97, 2009.

- [27] L. P. James, P. R. Mayeux, and J. A. Hinson. Acetaminophen-induced hepatotoxicity. *Drug metabolism and disposition*, 31(12):1499–1506, 2003.
- [28] J. J. Jones, J. E. Settle, R. M. Bond, C. J. Fariss, C. Marlow, and J. H. Fowler. Inferring tie strength from online directed behavior. *PloS one*, 8(1):e52168, 2013.
- [29] I. Kahanda and J. Neville. Using transactional information to predict link strength in online social networks. In *ICWSM*, 2009.
- [30] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.
- [31] D. Laney. 3d data management: Controlling data volume, velocity and variety. *META Group Research Note*, 6:70, 2001.
- [32] E. Leicht, P. Holme, and M. E. Newman. Vertex similarity in networks. *Physical Review E*, 73(2):026120, 2006.
- [33] D. Lin. Online learning of nonparametric mixture models via sequential variational approximation. In *Advances in Neural Information Processing Systems*, pages 395–403, 2013.
- [34] Z. Ma and A. Leijon. Bayesian estimation of beta mixture models with variational inference. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(11):2160–2173, 2011.
- [35] J. D. Mcauliffe and D. M. Blei. Supervised topic models. In *Advances in neural information processing systems*, pages 121–128, 2008.
- [36] A. K. McCallum. Mallet: A machine learning for language toolkit. <http://mallet.cs.umass.edu>, 2002.
- [37] D. Mimno, W. Li, and A. McCallum. Mixtures of hierarchical topics with pachinko allocation. In *Proceedings of the 24th international conference on Machine learning*, pages 633–640. ACM, 2007.
- [38] G. Moore. Cramming more components onto integrated circuits, *electronics*,(38) 8, 1965.
- [39] M. Newman. *Networks: an introduction*. Oxford University Press, 2010.
- [40] E. F. Nuwaysir, M. Bittner, J. Trent, J. C. Barrett, and C. A. Afshari. Microarrays and toxicology: the advent of toxicogenomics. *Molecular carcinogenesis*, 24(3):153–159, 1999.
- [41] J.-P. Onnela, J. Saramäki, J. Hyvönen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.-L. Barabási. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences*, 104(18):7332–7336, 2007.

- [42] N. R. Pal and J. C. Bezdek. On cluster validity for the fuzzy c-means model. *Fuzzy Systems, IEEE Transactions on*, 3(3):370–379, 1995.
- [43] J. K. Pritchard, M. Stephens, and P. Donnelly. Inference of population structure using multilocus genotype data. *Genetics*, 155(2):945–959, 2000.
- [44] S. Rogers, M. Girolami, C. Campbell, and R. Breitling. The latent process decomposition of cdna microarray data sets. *Computational Biology and Bioinformatics, IEEE/ACM Transactions on*, 2(2):143–156, 2005.
- [45] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 487–494. AUAI Press, 2004.
- [46] W. M. S. Russell, R. L. Burch, and C. W. Hume. The principles of humane experimental technique. 1959.
- [47] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown. Quantitative monitoring of gene expression patterns with a complementary dna microarray. *Science*, 270(5235):467–470, 1995.
- [48] J. Sethuraman. A constructive definition of dirichlet priors. Technical report, DTIC Document, 1991.
- [49] S. Sintos and P. Tsaparas. Using strong triadic closure to characterize ties in social networks. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1466–1475. ACM, 2014.
- [50] J. Tang, T. Lou, and J. Kleinberg. Inferring social ties across heterogenous networks. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 743–752. ACM, 2012.
- [51] W. Tang, H. Zhuang, and J. Tang. Learning to infer social ties in large networks. In *Machine Learning and Knowledge Discovery in Databases*, pages 381–397. Springer, 2011.
- [52] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476), 2006.
- [53] T. Uehara, Y. Minowa, Y. Morikawa, C. Kondo, T. Maruyama, I. Kato, N. Nakatsu, Y. Igarashi, A. Ono, H. Hayashi, et al. Prediction model of potential hepatocarcinogenicity of rat hepatocarcinogens using a large-scale toxicogenomics database. *Toxicology and applied pharmacology*, 255(3):297–306, 2011.
- [54] T. Uehara, A. Ono, T. Maruyama, I. Kato, H. Yamada, Y. Ohno, and T. Urushidani. The japanese toxicogenomics project: application of toxicogenomics. *Molecular nutrition & food research*, 54(2):218–227, 2010.



- [55] H. M. Wallach, D. M. Mimno, and A. McCallum. Rethinking lda: Why priors matter. In *Advances in neural information processing systems*, pages 1973–1981, 2009.
- [56] P. Workman, E. Aboagye, F. Balkwill, A. Balmain, G. Bruder, D. Chaplin, J. Double, J. Everitt, D. Farningham, M. Glennie, et al. Guidelines for the welfare and use of animals in cancer research. *British journal of cancer*, 102(11):1555–1577, 2010.
- [57] R. Xiang, J. Neville, and M. Rogati. Modeling relationship strength in online social networks. In *Proceedings of the 19th international conference on World wide web*, pages 981–990. ACM, 2010.
- [58] X. Xu, N. Yuruk, Z. Feng, and T. A. Schweiger. Scan: a structural clustering algorithm for networks. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 824–833. ACM, 2007.
- [59] J. Yin and J. Wang. A dirichlet multinomial mixture model-based approach for short text clustering. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 233–242. ACM, 2014.
- [60] K. Yu, B. Gong, M. Lee, Z. Liu, J. Xu, R. Perkins, and W. Tong. Discovering functional modules by topic modeling rna-seq based toxicogenomic data. *Chemical research in toxicology*, 27(9):1528–1536, 2014.