Bayesian Nonparametric Learning for Complicated Text Mining

Junyu Xuan

A thesis submitted for the Degree of
Doctor of Philosophy

Faculty of Engineering and Information Technology
University of Technology Sydney

October, 2016

CERTIFICATE OF AUTHORSHIP/ORGINALITY

This thesis is the result of a research candidature conducted jointly with another University as part of a collaborative Doctoral degree. I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of the requirements for a degree except as fully acknowledged within the text.

I also certify that this thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate

ACKNOWLEDGEMENTS

I would like to express my earnest thanks to my principal supervisor, Professor Jie Lu, and my co-supervisor, Professor Guangquan Zhang. Their comprehensive guidance has covered all aspects of my PhD study, including research methodology, research topic selection, experiments, academic writing skills and thesis writing, even sentence structure and formulas. Their critical comments and suggestions have strengthened my study significantly. Their strict academic attitude and respectful personalities have benefited my PhD study and will be a great treasure throughout my life. Without their excellent supervision and continuous encouragement, this research could not have been finished on time. Thanks to you both for your kind help! I would like to thank Dr. Richard Yi Da Xu for his advice on this study. Without his help it would not have been possible for me to plunge into this area and conduct this research.

I am grateful to all members of the Decision Systems and e-Service Intelligent (DeSI) Lab in the Centre for Quantum Computation and Intelligent Systems (QCIS) for their careful participation in my presentation and valuable comments on my research. I would like to thank Ms. Sue Felix and Jem Moore for helping me to proofread my publications and this thesis.

I wish to express my appreciation for the financial support I received for my study. Special thanks go to the China Scholarship Council (CSC) and the University of Technology Sydney (UTS).

Last but not least, I would like also to thank my family members. Thanks to my mother, father and sisters for their conscious encouragement and generous support.

ABSTRACT

Text mining has gained the ever-increasing attention of researchers in recent years because text is one of the most natural and easy ways to express human knowledge and opinions, and is therefore believed to have a variety of application scenarios and a potentially high commercial value. It is commonly accepted that Bayesian models with finite-dimensional probability distributions as building blocks, also known as parametric topic models, are effective tools for text mining. However, one problem in existing parametric topic models is that the hidden topic number needs to be fixed in advance. Determining an appropriate number is very difficult, and sometimes unrealistic, for many real-world applications and may lead to over-fitting or under-fitting issues. Bayesian nonparametric learning is a key approach for learning the number of mixtures in a mixture model (also called the model selection problem), and has emerged as an elegant way to handle a flexible number of topics. The core idea of Bayesian nonparametric models is to use stochastic processes as building blocks, instead of traditional fixed-dimensional probability distributions. Even though Bayesian nonparametric learning has gained considerable research attention and undergone rapid development, its ability to conduct complicated text mining tasks, such as: document-word co-clustering, document network learning, multi-label document learning, and so on, is still weak. Therefore, there is still a gap between the Bayesian nonparametric learning theory and complicated real-world text mining tasks.

To fill this gap, this research aims to develop a set of Bayesian nonparametric models to accomplish four selected complex text mining tasks. First, three Bayesian nonparametric sparse nonnegative matrix factorization models, based on two innovative dependent Indian buffet processes, are proposed for document-word co-clustering tasks. Second, a Dirichlet mixture probability measure strategy is proposed to link the topics from different layers, and is used to build a Bayesian nonparametric deep topic model for topic hierarchy learning. Third, the thesis develops a Bayesian nonparametric relational topic model for document network learning tasks by a subsampling Markov random field. Lastly, the thesis develops Bayesian nonparametric cooperative hierarchical structure models for multi-label document learning task based on two stochastic process operations: inheritance and cooperation. The findings of this research not only contribute to the development of Bayesian nonparametric learning theory, but also provide a set of effective tools for complicated text mining applications.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Background

Text mining, a variation on the field of data mining (Han et al., 2011) that tries to find interesting patterns from large datasets, refers generally to the process of extracting interesting and non-trivial information and knowledge from unstructured text (Gupta and Lehal, 2009). Since it is believed that the majority of information, approximately 80%, is stored as text, text mining has not only attracted scholars from the academic community but also industry practitioners. The text mining has gained ever-increasing attention by researchers in recent years, which makes it a very active research field. Many research directions have been explored in this field, such as text clustering (Aggarwal and Zhai, 2012), text classification (Lodhi et al., 2002), text summarisation (Goyal et al., 2013), information retrieval (William B. Frakes, 1992), and so on. Beyond dealing with simple document-word structures, more intricate situations have also been investigated, such as document network (e.g., paper citation networks or email reply networks) and document labels (e.g., authors or emotion tags). These more intricate situations bring tougher challenges to researchers because they significantly extend the application scenarios of this field. Given text is one of the most easiest and most natural ways for humans to express their knowledge and opinions, it is believed that text mining has a variety of application scenarios and a high-value commercial potential. For example, 1) Customer relationship management, which aims to manage the contents of clientsąŕ messages by automatically rerouting specific requests to the appropriate service, or supply immediate answers to the most frequently asked questions (Gupta and Lehal, 2009); 2) Online media service text mining, which is being used by large media companies, such as the Tribune Company, to clarify information and provide readers with a better search experience, which in turn increases the companyąŕs revenue[1]; 3) Academic analysis, where organisations and researchers might obtain the status and trends in a certain research field through analysing large amounts of published academic papers (Miner, 2012).

Topic models (Hofmann, 1999; Blei et al., 2003), which are Bayesian models with finite-dimensional probability distributions as building blocks, are commonly accepted as a very successful tool for text mining (Alghamdi and Alfalqi, 2015; Sun et al., 2012). These models normally comprise two stages: 1) estimate or compute the posterior distribution of the latent variables in a probabilistic model from a collection of text; 2) for new documents, answer the question at hand (e.g., classification and retrieval) via the learned posterior distribution of the latent variables (Blei, 2004). A set of significant latent variables that form interesting patterns within texts are termed as *topics*. Within different document collections, the subject of the topic will be different. If the academic papers of IEEE are taken as input documents, then a topic may be a research field: *computer vision*, *text mining* or *image segmentation*. If the emails in an organisation are taken as input documents, then the topics may be the concerns of its

[1]https://en.wikipedia.org/wiki/Text_mining

employees: *salary increases*, *heavy workloads* or *the end-of-year party*. If news webpages are taken as input documents, then the topic may be a news story: *Malaysia Airlines Flight 370 Disappears*, *Ebola Strikes West Africa* or *ISIS Declares an Islamic Caliphate* [2]. These topics can be seen as a summarisation or abstraction of the whole document collection. Discovering these hidden topics could improve the services of IEEE, such as the ability to search, browse or visualise academic papers; help an organisation understand and resolve the concerns of its employees; assist news websites to organise the large amounts of news webpages and provide online news services. Except for the traditional text mining tasks (e.g., text classification (Ramage et al., 2009) and text stream (Wang and McCallum, 2006)), there are existing some complicated text mining tasks, such as: 1) document-word co-clustering; 2) topic hierarchy learning. An extension of classical LDA, i.e., hierarchically labeled LDA (Perotte et al., 2011), is proposed to resolve this task; 3) document network learning. A link variable is introduced into LDA to model the document network (Chang and Blei, 2009; Chang et al., 2010; Chen, Zhu, Xia and Zhang, 2015); 4) multi-label document learning. The labels of documents could be author (Rosen-Zvi et al., 2004; Steyvers et al., 2004; Rosen-Zvi et al., 2010) or emotional tags (Lin and He, 2009). Based on the existing parametric topic models, these complicated text mining tasks could be resolve in a degree.

A major issue in existing parametric topic models for text mining is that the hidden topic number in the defined priors needs to be fixed in advance. This number is normally chosen with domain knowledge. After fixing the number of topics, Dirichlet, multinomial, and other fixed-dimensional distributions can be adopted as the building blocks for (parametric) topic models. However, determining an appropriate number is very difficult, and sometimes unrealistic, in

[2]http://www.theatlantic.com/international/archive/2014/12/the-10-biggest-international-stories-of-2014/383935/

many real-world applications. For example, limiting each document to a fixed exact number of topics is unrealistic for any given corpus. Furthermore, predetermining a number of topics may lead to overfitting when there are too many, such that relatively specific topics will not generalise well to unseen observations; Underfitting is the opposite case. When there are too few topics, unrelated observations will be assigned to the same topic (Dai and Storkey, 2015). A number of other methods can be used to nominate the number of topics, including cross-validation techniques (Griffiths and Steyvers, 2004), but they are slow because the algorithm has to be restarted a number of times before determining the optimal number of topics (Griffiths and Steyvers, 2004; Dai and Storkey, 2015).

One elegant approach to resolve the above issue is *Bayesian nonparametric learning* - a key approach for learning the number of mixtures in a mixture model, also called the model selection problem (Gershman and Blei, 2012). This number is supposed to be inferred from the data, i.e., let the data speak. The idea of Bayesian nonparametric learning is to use stochastic processes (i.e., Dirichlet process (Ghosal, 2010), multinomial process, beta process (Tamara Broderick and Pitman, 2012), poisson process (Iwata et al., 2013)) to replace traditional fixed-dimensional probability distributions. The merit of these stochastic processes is that they have a theoretically infinite number factors, and let the data determine the number of used factors. Many probabilistic models with fixed dimensions have been extended into infinite ones with the help of stochastic processes. One example is the famous Gaussian mixture model, which was extended into an infinite Gaussian mixture model (Rasmussen, 1999) using a Dirichlet process. Another famous example is the hidden Markov model (Rabiner, 1989; Yu, 2010) that has been extended into an infinite hidden Markov model (Wulsin et al., 2014) using hierarchical Dirichlet processes (Teh, 2006). The core of the

Bayesian nonparametric learning is to build or infer the model for a particular problem using stochastic processes as its building blocks.

Although the Bayesian nonparametric learning has experienced a boom in recent years, there is still a huge gap between existing work and complicated text mining tasks, such as: 1) document-word co-clustering; 2) topic hierarchy learning; 3) document network learning; 4) multi-label document learning. Accomplishing all these complex tasks requires the further development new technologies and theories. This thesis intends to fill that gap by developing a set of Bayesian nonparametric models for these text mining tasks.

1.2 Research questions and objectives

This research aims to develop a set of Bayesian nonparametric models for four selected text mining tasks and will answer the following research questions:

QUESTION 1 *How to co-cluster documents and words under Bayesian nonparametric setting?*

A text corpus is a form of dyadic data normally represented as a matrix with rows and columns representing two entities (i.e., document and word). An important text mining task pertinent to a text corpus is to cluster each entity. For example, document clustering could benefit the efficiency of an organisation or the browsing of a corpus; and word clustering could be used for automatic construction of a statistical thesaurus or the enhancement of queries (Crouch, 1988). However, clustering algorithms that cluster documents and words separately do not perform well on such problems because the relationship between documents and words is overlooked. By comparison, co-clustering (Dhillon, 2001; Bichot, 2010), where documents and words are clustered at the same time, can achieve better performance by investigating the relationship between documents and words (Shan and Banerjee, 2008).

QUESTION 2 *How to learn a topic hierarchy from documents under Bayesian nonparametric setting?*

The topics within a corpus normally have different granularities. For example, *image segmentation* is a sub-area of the research field *computer vision*, therefore *computer vision* is more general than *image segmentation*. Obtaining different levels of summarisation or abstraction in a document collection, called topic hierarchy, is of practical benefit to many real-world applications, including, but not limited to, document retrieval, summarisation, browsing and visualisation. Take information retrieval (Fersini et al., 2008) as an example. A user could start with a more general description about information that she wants, and could gradually choose more specific topics provided by the retrieval system according to the learned topic hierarchy until the desired information is found.

QUESTION 3 *How to learn from a document network under Bayesian nonparametric setting?*

Sometimes, there are links between the documents in a corpus. A paper citation network (Guo et al., 2014) is an example of a document network in which the academic papers are linked by their citation relations; an email network (Klimt and Yang, 2004) is a document network in which the emails are linked by their reply relations; a webpage network (Park, 2003) is a document network in which webpages are linked by their hyperlinks. Since these links also express the nature of the documents, it is apparent that hidden topic discovery should consider these links as well. Learning from the document network could support relevant scientific papers or web pages recommendation.

QUESTION 4 *How to learn from the multi-label documents under Bayesian nonparametric setting?*

Multi-label learning (Gibaja and Ventura, 2015; Zhang and Zhou, 2014) is a significant learning paradigm in which each instance may be assigned with more than one label. For instance, each academic paper may have more than one author, and learning from this data could help to identify the academic interests of authors and recommend potential collaborators according their interests (Rosen-Zvi et al., 2004; Xuan, Lu, Zhang, Xu and Luo, 2015); a patent may be associated with several categories, and automatically assign large amount of new patents to correct categories could save the costs in human resources and time (Cong and Tong, 2008).

This research aims to achieve the following objectives, which are expected to answer the above research questions:

OBJECTIVE 1 *To develop a Bayesian Nonparametric Sparse Nonnegative Matrix Factorization Model.*

This objective corresponds to research Question 1. Nonnegative matrix factorisation (NMF) aims to factorise a matrix into two optimised nonnegative factor matrices, appropriate for its intended application, which could benefit many tasks, including document-word co-clustering the weighting of a matrix papers by its words). However, traditional NMF typically assumes the number of latent factors (i.e., the dimensionality of the factor matrices) to be fixed. This study will develop a nonparametric sparse NMF model to remove this assumption and learn the correlation between documents and words.

OBJECTIVE 2 *To develop a Bayesian Nonparametric Deep Topic Model.*

This objective corresponds to research Question 2. Traditional topic models only output a topic set without its hierarchical structure. Some extensions have been made to learn the topic hierarchy based on topic models, such as tree labeled LDA (Slutsky et al., 2013), hierarchically labeled LDA (Perotte et al., 2011), Pachinko allocation (Li and McCallum, 2006; Mimno et al., 2007) and fixed structure LDA (Reisinger and Paşca, 2009). However, these extensions suffer from one drawback: the number of topics need to be prefixed. This study will develop a new Dirichlet mixture probability measurement strategy to facilitate topic hierarchy learning without the need to predefine the number of topics.

OBJECTIVE 3 *To develop a Bayesian Nonparametric Relational Topic Model.*

This objective corresponds to research Question 3. Similar studies focusing on the hidden topics discovered from a document network using some relational topic models (RTM) (Chang and Blei, 2009; Chang et al., 2010; Chen, Zhu, Xia and Zhang, 2015) have already been successfully developed. Unlike traditional topic models (Blei et al., 2003; Blei, 2012) that focus on mining the hidden topics from a document corpus (without links between documents), RTMs can force discovered topics to inherit their document network structure. One drawback of existing RTMs is that they are built with fixed-dimensional probability distributions, such as Dirichlet, multinomial, gamma and possion distributions, which require their dimensions to be fixed before use. This study will develop a Bayesian nonparametric relational topic model based on stochastic processes instead of probability distributions.

OBJECTIVE 4 *To develop Bayesian Nonparametric Cooperative Hierarchical Structure Models.*

This objective corresponds to research Question 4. Existing studies in Bayesian nonparametric learning area mainly focus on non-cooperative hierarchical structures based on hierarchical Dirichlet processes (HDP) (Teh, 2006) or gamma-negative binomial processes (Zhou and Carin, 2015) and their variations (Dai and Storkey, 2015; Paisley et al., 2015; Canini and Griffiths, 2011). To capture the cooperative hierarchical Structure, this study will develop cooperative hierarchical Dirichlet processes and mixed gamma-negative binomial processes based on two designed process operations: inheritance and cooperation.

1.3 Research contributions

The main contributions of this study are concisely summarised as follows:

- Two new dependent Indian buffet processes (i.e., bivariate Beta distribution-based dIBP and copula-based dIBP) with simpler model structures and more direct correlation modeling capabilities have been developed as alternatives to the existing GP-based dIBP;
- Three dependent Indian buffet processes-based Bayesian nonparametric doubly sparse nonnegative matrix factorisation models are proposed to remove the assumption of the traditional NMF that the number of factors needs to be fixed in advance;
- A new Dirichlet mixture probability measure is designed to facilitate flexible topic hierarchy when learning from a document collection, while two sampling inference algorithms, i.e., a truncated version and a slice version, are proposed to facilitate the posterior inference for the proposed model;
- Cooperative hierarchical Dirichlet process and mixed gamma-negative binomial process are innovatively proposed based on two operations on ran-

dom measures: *Inheritance* and *Cooperation*, which could be used to model the cooperative hierarchical structures that cannot be modelled by existing Bayesian nonparametric models;

- two constructive representations (i.e., stick-breaking and the international restaurant process) for cooperative hierarchical Dirichlet process are proposed to facilitate the model inference, which rises to the challenge brought about by *inheritance* and *cooperation* between random measures;

- The expectation of component numbers from cooperative hierarchical Dirichlet processes and mixed Gamma-negative binomial processes are theoretically and empirically analysed to understand the behavior and sensitivity of the processes under different parameters.

1.4 Research significance

The theoretical and practical significance of this research is summarized as follows:

Theoretical significance: The findings of this study contribute to the Bayesian nonparametric learning community in the following two ways: enriching the theoretical analysis methodology for Bayesian nonparametric learning; and facilitating the application of Bayesian nonparametric learning to more complex scenarios. More specifically, as the core of the Bayesian nonparametric learning is the manipulation of stochastic processes, the proposed ideas, such as: dependent Indian buffet processes, Dirichlet mixture probability measure, and cooperation between Dirichlet processes or gamma processes, could enrich the approach for powerful and meaningful manipulations on stochastic processes which lay the foundation for Bayesian nonparametric learning area. Meanwhile, the proposed ideas on the stochastic processes manipulation are particularly designed for more complex scenario modeling, such as: modeling the correlation between documents and words, modeling document networks, modeling multilabels of documents, and so on. The extended application scenarios could not only improve the impact of this area, but also motivate its continuing development.

Practical significance: The findings of this study contribute to the benefit of society given the important role text plays in modern life. The findings help resolve the real-world problems of corporations and organisations and improve quality of life for individuals. Corporations can manage their customer relationships better, or improve their product designs by analysing feedback and comments from their customers. Organisations can understand their employees concerns by analysing internal email. Analysis of an individualąŕs search history can help search engines improve user experience and satisfaction. There is potential for many other such applications that could benefit from this study.

1.5 Thesis structure

The structure of the thesis is shown in Figure 1.1 and the chapters are organized as follows:

- In Chapter 2, this thesis reviews the current state-of-the-art of research on Bayesian nonparametric learning with a particular focus on text mining, including its basic concepts, commonly used mechanisms, existing models, inference strategies, and application scenarios.

- In Chapter 3, this thesis develops Bayesian nonparametric sparse nonnegative matrix factorisation models for document-word co-clustering. Three strategies, which induce three implementations for the Bayesian nonparametric sparse nonnegative matrix factorisation model, are proposed to cor-

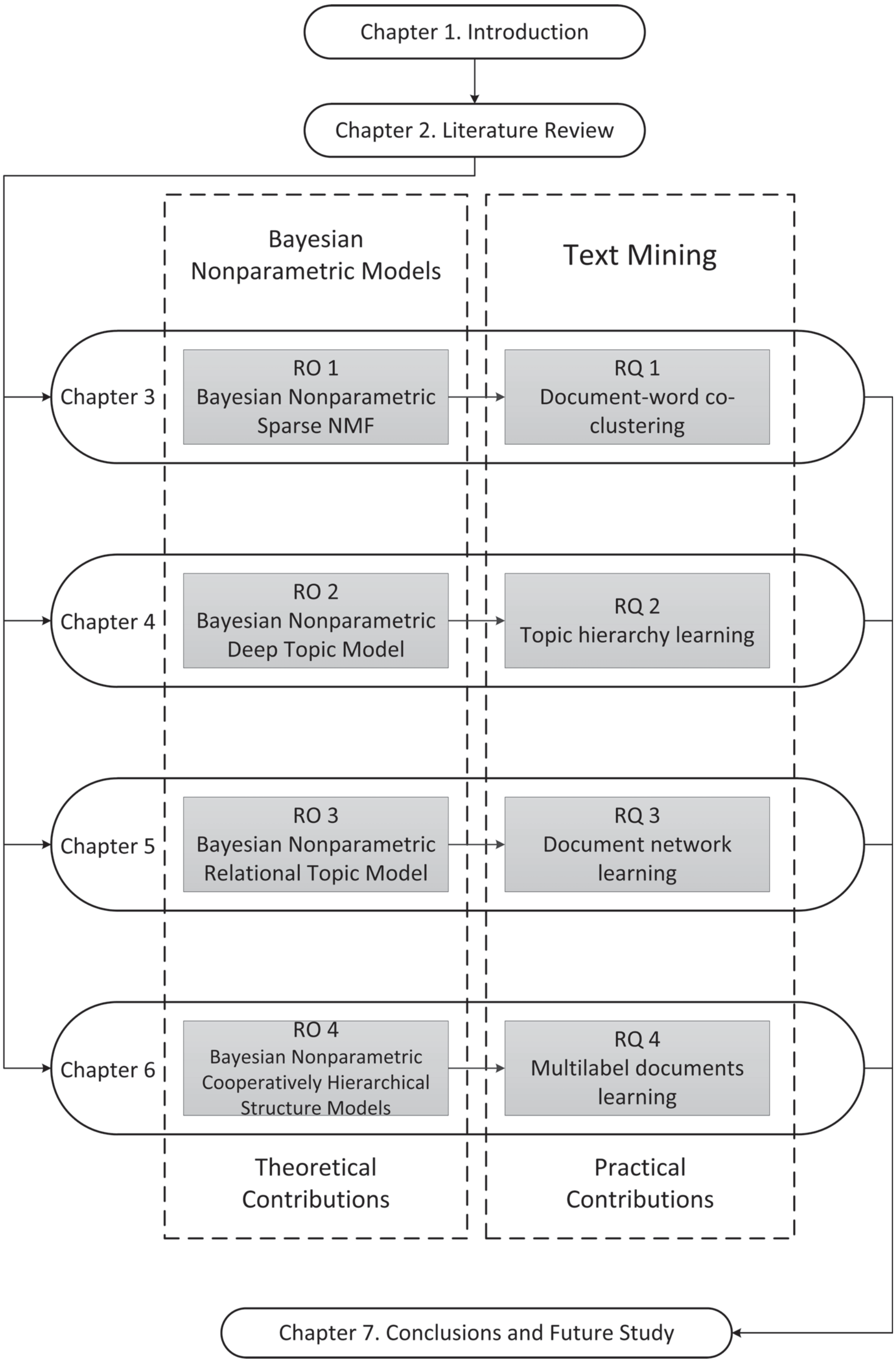


Figure 1.1: Thesis structure

relate documents and words, and are theoretically and empirically compared.

- In Chapter 4, this thesis develops a Bayesian nonparametric deep topic model for topic hierarchy learning. Although the input data is same as the model in Chapter 3, this model can learn a topic hierarchy rather than a set of topics, as in Chapter 3. A Dirichlet mixture probability measure is presented to link the topics located in different layers.

- In Chapter 5, this thesis presents a Bayesian nonparametric relational topic model for document network learning. As opposed to the above two Chapters, this Chapter focuses on text corpora with document links. A strategy is presented to incorporate the network structure into the Bayesian nonparametric model design and inference. This model could be applied to document link prediction.

- In Chapter 6, this thesis presents two Bayesian nonparametric cooperative hierarchical structure models for multi-label document learning. This Chapter focuses on text corpora with multiple labels. Two operations between the stochastic processes are formally defined and used for cooperative hierarchical structure modeling, including two implementations: cooperative hierarchial Dirichlet processes and mixed gamma-negative binomial processes. These models are characterised by the abilities in the supervised document learning.

- Finally, Chapter 7 summarises the findings of the thesis and points to directions for future work.

1.6 Publications related to this thesis

Below is a list of the refereed international journal and conference papers during my PhD research that have been published or currently under review:

Published:

Journals :

1. **Junyu Xuan**, Jie Lu, Guangquan Zhang, and Xiangfeng Luo, *Topic models for graph mining*, IEEE Transactions on Cybernetices, Vol.45, No.12, pp.2792-2803, 2014. (ERA rank: A)
2. Xiangfeng Luo, **Junyu Xuan**, Guangquan Zhang, and Jie Lu, *Measuring the semantic uncertainty of news events for evolution potential estimation*, ACM Transactions on Information Systems, Vol.34, No.4, 2016. (ERA rank: A)
3. **Junyu Xuan**, Xiangfeng Luo, Jie Lu, and Guangquan Zhang, *Uncertainty analysis for the keyword system of web events*, IEEE Transactions on Systems, Man and Cybernetics: Systems, Vol.46, No.6, pp.829-842, 2016. (ERA rank: B)

Conferences :

4. **Junyu Xuan**, Jie Lu, Guangquan Zhang, Richard Yida Xu, and Xiangfeng Luo, *Infinite author topic model through mixed Gamma negative binomial processes*, IEEE International Conference on Data Mining (ICDM), Atlantic City, USA, 2015. (ERA rank: A)
5. **Junyu Xuan**, Jie Lu, Guangquan Zhang, and Xiangfeng Luo, *Release 'bag-of-words' assumption of latent Dirichlet allocation*, International Conference on Intelligent Systems and Knowledge Engineering

(ISKE), Shenzhen, China, pp.83-92, 2014. (ERA rank: B, Best Student Paper Award)

6. **Junyu Xuan**, Jie Lu, Guangquan Zhang, and Xiangfeng Luo, *Extension of similarity measures in VSM: from orthogonal coordinate system to affine coordinate system*, The 2014 International Joint Conference on Neural Networks (IJCNN), Beijing, China, pp.4084-4091, 2014. (ERA rank: A)

Under review:

1. **Junyu Xuan**, Jie Lu, Guangquan Zhang, Richard Yida Xu, and Xiangfeng Luo, *Nonparametric relational topic model through dependent Gamma processes*, IEEE Transactions on Knowledge and Data Engineering, under review.

2. **Junyu Xuan**, Jie Lu, Guangquan Zhang, Richard Yida Xu, and Xiangfeng Luo, *Dependent Indian buffet process-based sparse nonparametric nonnegative matrix factorization*, IEEE Transactions on Neural Network and Learning Systems, under review.

3. **Junyu Xuan**, Jie Lu, Guangquan Zhang, Richard Yida Xu, and Xiangfeng Luo, *Cooperatively hierarchical Dirichlet processes*, Artificial Intelligence, under review.

4. **Junyu Xuan**, Jie Lu, Guangquan Zhang, Richard Yida Xu, and Xiangfeng Luo, *A Bayesian nonparametric model for multi-label learning*, Machine Learning, under review.

Chapter 2

Literature Review

This chapter reviews the existing researches on Bayesian nonparametric learning (BNL) in detail. In Section 2.1, the definition and motivation of Bayesian nonparametric learning are introduced. Section 2.2 presents the basic ingredients of this area followed by their manipulations summarized in Section 2.3. Section 2.4 and 2.5 discuss the applications of Bayesian nonparametric learning on two settings: supervised learning and tree structure learning. Section 2.6 collects the nonparametric extensions of existing classical (parametric) models. Section 2.7 introduces the inference techniques used in this community. Section 2.8 describes the domains where Bayesian nonparametric learning has been applied. Section 2.9 briefly summarizes this chapter.

2.1 Definitions of Bayesian nonparametric learning

Many machine learning tasks involve building a model class defined by a set of parameters and then inferring an (globally or locally) optimized set of parameters. The model parameters control the goodness-of-fit to the data and the model complexity, so inappropriate model parameters may lead to underfitting (i.e., bad goodness-of-fit but good model complexity) or overfitting (i.e., good goodness-of-fit but bad model complexity) problems. To determine or learn these parameters is termed model selection or model adaptation problem. Traditional methods for the model selection are based on human labor or restated several times to find an optimal setting, which are quite time-consuming and not scalable to large unfamiliar data. Fortunately, Bayesian nonparametric learning has emerged as an elegant way to resolve the above problem.

One definition of Bayesian nonparametric models comes from *Encyclopedia of Machine Learning* (Orbanz and Teh, 2010) as

Definition 1 *A Bayesian nonparametric model (BNP) is a Bayesian model on an infinite-dimensional parameter space. The parameter space is typically chosen as the set of all possible solutions for a given learning problem.*

The ability to build models on infinite-dimensional parameter spaces is due to stochastic processes as the alternative of probabilistic distributions. One major contribution of Bayesian nonparametric learning to the machine learning community is to introduce stochastic processes for two-dimensional space partition rather than intuitively time series events modeling. Therefore, an alternative definition for Bayesian nonparametric learning is given as follow

Definition 2 *Bayesian nonparametric learning (BNL) is to build and inference the probabilistic models for specific data structures or tasks based on stochastic processes and their manipulations.*

From the above definition, it can be seen that the core of BNL is to manipulate stochastic processes according to the target task. Therefore, the stochastic processes currently used in this area is reviewed with their properties in the next section followed by developed manipulations on them.

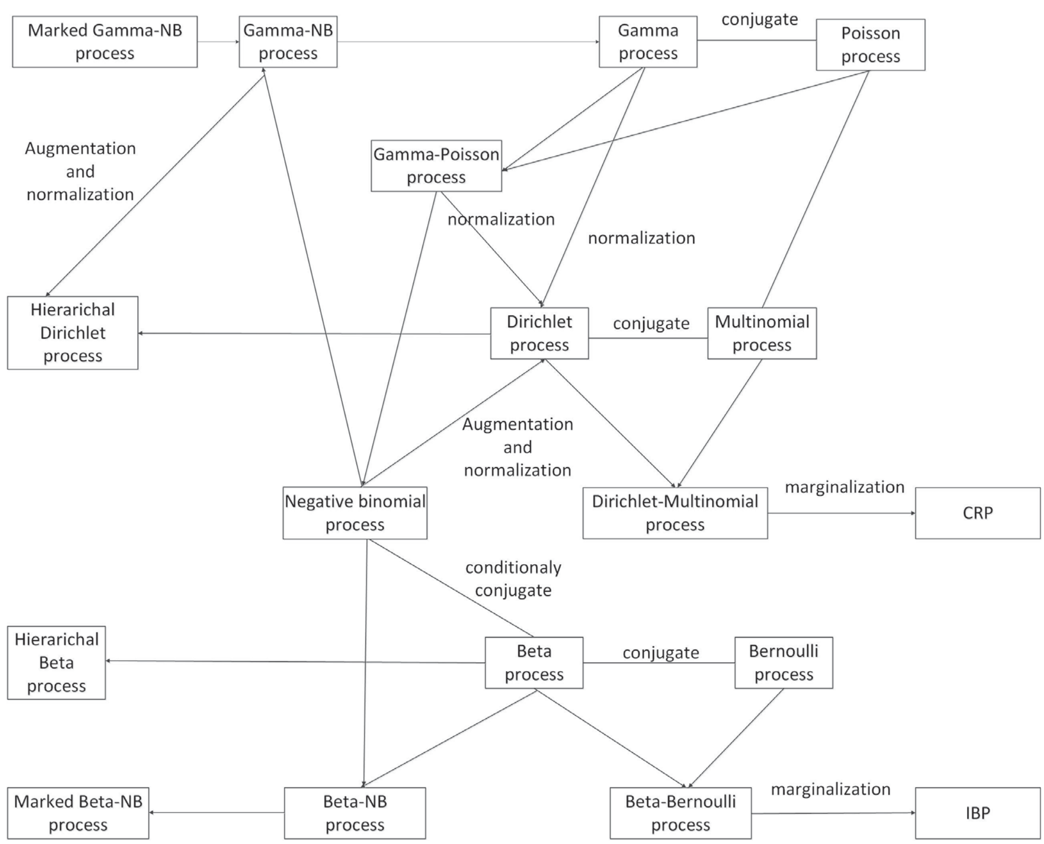


Figure 2.1: The relations between stochastic processes.

2.2 Basic ingredients: stochastic processes

In probability theory, a stochastic process is a collection of random variables. According to different properties of these random variables, there are a number of different stochastic processes defined in the literature. In the following, only the ones commonly adopted in BNL for the machine learning community are reviewed. Some important properties are summarized in Table 2.1, and their relations have been illustrated in Fig. 2.1.

— *Poisson Process (PP).* (Kingman, 1992) A poisson process $\Pi \sim PP(G_0)$ on the product space $\mathbb{Z}^+ \times \Theta$, where G_0 is a base measure over Θ. For any subset $A \subset \Theta$, $\Pi(\Omega) \sim Poisson(G_0(\Omega))$ and $\{\Pi(\Omega_i)\}_{i=1}^N$ are independent with each other for any disjoint subsets $\{\Omega_i\}_{i=1}^{N}$. The Levy measure of PP is

$$\nu(\mathrm{d}\theta, \mathrm{d}\lambda) = \delta_1(\mathrm{d}\lambda) G_0(\mathrm{d}\theta) \tag{2.1}$$

where $\delta_1(\mathrm{d}\lambda)$ is a unit point mass at $\lambda = 1$. An important property of PP is the equivalence with completely random measures (CRM) (i.e., any draw from a PP is a completely random measure) (Kingman, 1967). This property places it at the core of BNL. A number of stochastic processes in this area have been proofed as special cases of CRM, and the manipulations of these stochastic processes thank to the theorems of PP which will be discussed in the following section.

— *Dirichlet Process (DP).* (Ferguson, 1973; Teh, 2010) DP is the pioneer and foundation of the BNL. Its definition is as follow: A DP, which is specified by a base measure H on a measurable space Θ and a concentration parameter α, is a set of countably infinite random variables that can be seen as a (probability) measure on partitions from a random infinite partition $\{\Omega\}_{k=1}^{\infty}$ of Θ. An explicit representation for DP by stick-breaking process (Teh, 2010) is

$$G = \sum_k \pi_k \delta_{\theta_k} \tag{2.2}$$

where $\sum_k \pi_k = 1$. One property of DP is that: for any finite partition $\{\Omega\}_{k=1}^{K}$, the variables (measures on these partitions) from DP satisfy a Dirichlet distribution parameterized by the measures from based measure H on relative areas

$$(G(\Omega_1), G(\Omega_2), \ldots, G(\Omega_K)) \sim Dir(\alpha H(\Omega_1), \alpha H(\Omega_2), \ldots, \alpha H(\Omega_K)) \tag{2.3}$$

where G is a realization of $DP(\alpha, H)$ and $Dir()$ denotes the Dirichlet distribution. Considering the property $\sum_k \pi_k = 1$ of DP, it is normally used as the mixing distribution in mixture models, which makes it the most active and successful stochastic process in BNL.

— *Beta Process (BP).* (Hjort, 1990; Thibaux and Jordan, 2007) A beta process $B \sim BP(c, B_0)$ is a positive Levy process whose Levy measure depends on two parameters: c is a positive concentration parameter and B_0 is a base measure on Θ. If B_0 is continuous, the Levy measure on $\Theta \times [0, 1]$ of the BP is

$$\nu(\mathrm{d}\theta, \mathrm{d}p) = cp^{-1}(1-p)^{c-1}\mathrm{d}pB_0(\mathrm{d}\theta) \tag{2.4}$$

A draw B is a set of points $(\theta_k, p_k) \in \Theta \times [0, 1]$ from a PP with base measure ν and

$$B = \sum_k p_k \delta_{\theta_k} \tag{2.5}$$

where δ_{θ_k} is a unit point mass at θ_k. A draw from BP is a list of (countablely infinite) probabilities (different from DP, the summation does not need to be one) which is commonly used as the prior for the rows of a matrix in factor analysis (Tamara Broderick and Pitman, 2012). A three-parameterized version is proposed in (Broderick et al., 2015).

— *Bernoulli Process (BeP).* (Kingman, 1992) Let B be a measure on Ω. Define a Bernoulli process with hazard measure B, written $X \sim BeP(B)$, as the Levy process with Levy measure,

$$\nu(\mathrm{d}p, \mathrm{d}\omega) = \delta(\mathrm{d}p)B(\mathrm{d}\omega) \tag{2.6}$$

In summary, a Bernoulli process is similar to a PP, except that it gives weight 1 to points. BeP is normally used as the likelihood when BP is set as the prior.

— *Gamma Process (GaP).* (Roychowdhury and Kulis, 2015) A gamma process $\Gamma \sim GaP(c, \Gamma_0)$ is a positive Levy process whose Levy measure depends on two parameters: c is concentration parameter and G_0 is the base measure. The corresponding Levy measure on $\Theta \times \mathbb{R}^+$ of GaP is

$$\nu(d\theta, dr) = cr^{-1}e^{-cr}\mathrm{d}rG_0(\mathrm{d}\theta) \tag{2.7}$$

A draw Γ is a set of points $(\theta_k, r_k) \in \Theta \times \mathbb{R}^+$ from a PP with base measure ν and

$$\Gamma = \sum_k r_k \delta_{\theta_k} \tag{2.8}$$

Analogous to the relationship between Gamma and Dirichlet distributions, DP could be seen as a normalized GaP, i.e., $\pi_k = \frac{r_k}{\sum_j r_j}$. For the computational convenience, PP is usually selected as the likelihood when GaP is used as prior.

— *Negative-binomial Process (NBP).* (Zhou and Carin, 2012; Zhou et al., 2012) A NBP $I \sim NBP(r, p)$ is parameterized by two parts: r is a shape measure and p is the probability measure. It can also be represented as

$$X = \sum_k \kappa_k \delta_{\theta_k}, \kappa_k \sim NB(r_k, p_k) \tag{2.9}$$

NBP is an alternative as the likelihood for GaP prior. Comparing with PP, NBP has better variance-to-mean ratio and overdispersion level than PP for overdispersed count data (Zhou and Carin, 2015). Another prior for NBP is the BP for p and it is proofed that NBP is conjugate with three-parameterized BP (Broderick et al., 2015).

— *Others.* Except for the above processes, there exists other processes adopted for BNL, such as *Gumbel Process (GuP)* (Maddison et al., 2014), *Gaussian*

Process (GP) (Seeger, 2004), and Mondrian Process (MoP) (Roy and Teh, 2008; Balog and Teh, 2015). Since they are a little far from this study, they are not given more details here.

2.3 Manipulations of stochastic processes

The aim of the BNL is to utilize the above stochastic processes to model the data and then resolve the tasks. However, the modeling ability of any single stochastic process is very limited, there are, therefore, a number of manipulations on two or more stochastic processes developed in the literature account for more complicated situations. This thesis summarises the existing manipulations in the following.

2.3.1 Marginalization

According to de Finetti's Theorem (Aldous, 1985), an exchangeable sequences could be obtained through marginalizing Finetti mixing measure out as follows,

$$P(X_1, X_2, \cdots, X_n) = \int \prod_{i=1}^{n} P(X_i|\theta)P(\mathrm{d}\theta) \tag{2.10}$$

where $P(\cdot|\theta)$ is named mixture measure and $P(\mathrm{d}\theta)$ is named mixing measure. For the BNL, the mixing measure is from a stochastic process, so marginalizing out a stochastic process could obtain a new exchangeable measure. The two examples of this marginalization manipulation are shown below.

— *Chinese Restaurant Process (CRP).* (Aldous, 1985) CRP is an example of marginalization of DP through,

$$CRP(X) = \int P(X|G)\mathrm{d}G, \quad G \sim DP \tag{2.11}$$

Table 2.1: Properties of stochastic processes. ‘Stick-breaking’ denotes if the process has a stick-breaking constructive representation; ‘CRM’ denotes if the process is a completely random measure; ‘NRM’ denotes if the process is a normalized random measure; ‘Power-law’ denotes if there is a version with power-law (cluster number to data number) phenomenon.

Stochastic Process	Stick-breaking	CRM	NRM	Power-law
PP		$\surd$		
DP	$\surd$ (Teh, 2010)		$\surd$	PYP (Teh, 2006)
BP	$\surd$ (Tamara Broderick and Pitman, 2012)	$\surd$		
BeP		$\surd$		
GaP	$\surd$ (Roychowdhury and Kulis, 2015)	$\surd$		
NBP		$\surd$	$\surd$	BNBP (with BP as based measure) (Broderick et al., 2015)
HDP	$\surd$ (Wang et al., 2011)		$\surd$	HPYP (Blunsom and Cohn, 2011)
IBP	$\surd$ (Teh, Görür and Ghahramani, 2007)			

where mixing measure G is from a DP. CRP could be seen as a random partition of data points and its name comes from a metaphor for understanding this process. It is normally used as a marginalized representation for DP.

— *Indian Buffet Process (IBP).* Another example is IBP which is a marginalization of BP through,

$$IBP(X) = \int P(X|B)\mathrm{d}B, \quad B \sim BP \tag{2.12}$$

where mixing measure B is from a BP. The basic IBP is proposed in (Griffiths and Ghahramani, 2005, 2011). Its widespread popularity is due to its power to generate a binary matrix with infinite columns, especially suitable for the factor analysis.

2.3.2 Layering

The base measure of a stochastic process could be a continuous or discrete measure while a realization from a stochastic process is also a random measure. It is apparently possible to set a random measure from a stochastic process as the base measure for another stochastic process, and this natural idea could enable the statistical strength sharing between the random measures at the lower layer. An illustrative example is document modeling. The upper layer random measure could be understood as a topic pool and a lower measure for a document could be seen as the topics in this document inherited from the topic pool. Existing BNMs based on this manipulation are summarized as follow.

— *Hierarchical Dirichlet Process (HDP).* (Teh, 2006) HDP is built by piling a DP above another DP, which defines a distribution over a set of random (probability) measures $\{G_d\}_{d=1}^{D}$ and a global random probability measure G_0 over

Θ. The global measure G_0 is distributed as a DP parameterized by a concentration parameter α and a base (probability) measure H.

$$G_d \sim DP(\alpha_d, G_0), \quad G_0 \sim DP(\alpha, H) \tag{2.13}$$

Meanwhile, each random measure, which G_d is conditionally independent with others given G_0, is also distributed as a DP with parameter α_d and a base probability measure G_0. HDP is almost the first and most successful model using this manipulation so there are a lot of extensions developed based on this model, including the supervised Dai and Storkey (2015), incremental Gao et al. (2011), nested Paisley et al. (2015), adapted Zheng et al. (2014), tree Canini and Griffiths (2011), evolutionary Xu et al. (2008), and dynamic versions Ren et al. (2008).

— *Hierarchial Beta Process (HBP).* (Thibaux and Jordan, 2007) Akin to HDP, HBP is proposed so that the different BPs could share a common base discrete measure from a global BP through

$$B_d \sim BP(c_d, B_0), \quad B_0 \sim BP(c, H) \tag{2.14}$$

Since BP is an effective tool for the factor analysis, HBP could enable the factor sharing during the factor analysis, so it has been successfully used for document classification, convolutional factor analysis (Chen et al., 2011), shared subspace learning (Gupta et al., 2012), and image interpolation and denoising (Zhou et al., 2011).

— *Gamma-Negative Binomial Process (GNBP).* Different from the above, i.e., HDP and HBP, GNBP is not composed by single kind of processes (e.g., DP and BP). GNBP (Zhou and Carin, 2015) is composed by two kinds of processes (i.e., GaP and NBP) as follow

$$I_d \sim NBP(p, \Gamma_0), \quad \Gamma_0 \sim GaP(c, H) \tag{2.15}$$

More layers could be achieved by further assigning random measures to H from another GaP.

— *Beta-Negative Binomial Process (BNBP).* Analogous to negative binomial distribution, there are two parameters for a BNP. While GNBP places a GaP to one parameter, another layering manipulation BNBP (Broderick et al., 2015) is to place a BP on another parameter as follow

$$I_d \sim NBP(B_0, r), \quad B_0 \sim BP(c, H) \tag{2.16}$$

2.3.3 Superposition

Superposition is to combine two or more random measures together, like 'plus' operation. If the layering is seen as a *vertical* manipulation, the superposition here could be seen as a *horizontal* manipulation.

— *Superposition of PP.* According to Superposition Theorem in (Kingman, 1992), it is known that combining a set of independent Poisson processes yields a new Poisson process whose mean measure is the sum of mean measures of the individual ones as follows

$$\Pi_1 \oplus \cdots \Pi_n \sim PP(\mu_1 + \cdots + \mu_n) \tag{2.17}$$

where $\Pi_n \sim PP(\mu_n)$. Considering the relationship between PP and completely random measures, this theorem is the foundation for the superposition of other processes.

— *Superposition of DP.* Inspired by the superposition of PP, the superposition of DP is proposed in (Lin et al., 2010) as follows,

$$\begin{aligned} (c_1, c_2, \cdots, c_n) &\sim Dir(\mu_1(\Omega), \mu_2(\Omega), \cdots, \mu_n(\Omega)) \\ c_1G_1 \oplus c_2G_2 \oplus \cdots \oplus c_nG_n &\sim DP(\mu_1 + \mu_2 + \cdots + \mu_n) \end{aligned} \tag{2.18}$$

where $\{G\}$ are a set of independent DPs.

— *Superposition of normalized random measures (SNRM).* DP is one case of NRM. The more general superposition manipulation(Chen et al., 2012) is proposed as follow: The superposition of n independent normalized random measures $\{\mu_j\}_{j=1}^n$ on Θ is as follow

$$\mu_1 \oplus \cdots \oplus \mu_n = c_1\mu_1 + \cdots + c_n\mu_n \tag{2.19}$$

where $c_n = \frac{\mu_n(\Theta)}{\sum_j \mu_j(\Theta)}$. This more general definition extends this manipulation to more situations.

2.3.4 Subsampling

Subsampling is to (randomly) select part of infinite components in a random measure from a stochastic process. When certain conditions are satisfied, the selected components form a new random measure from an underlying process.

— *Thinned PP.* (Lin et al., 2010) Based on the Marking Theorem (Kingman, 1992), let $\Pi \sim PP(\mu)$ be a PP on the space Θ, and $q : \Theta \to [0, 1]$ be a measurable function. If independently drawing $z_\theta \in \{0, 1\}$ for each $\theta \in \Pi$ with $P(z_\theta = 1) = q(\theta)$, we can obtain a new PP as

$$\Pi_t = \{\theta \in \Pi : z_\theta = 1\} \sim PP(q\mu) \tag{2.20}$$

where Π_t still satisfies a PP on Θ with mean measure $q\mu$.

— *Thinned Completely Random Measures (TCRM).* (Foti et al., 2013) Let $\Pi = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}$ be a CRM on Θ and $q : \Theta \to [0, 1]$ be a measurable function. For each point (θ_k, π_k), define a bernoulli variable r_k with $P(r_k = 1) = q(\theta_k)$ (independent with other $\{r\}$).

$$TCRM = \sum_{k=1}^{\infty} r_k \pi_k \delta_{\theta_k} \tag{2.21}$$

TCRM is still a CRM with mean measure $q\mu$. Note that PP is a special case of CRM. TCRM generalizes this manipulation to all the CRM, such as BP and NBP.

— *Thinned DP.* (Lin et al., 2010) Let $G \sim DP(\mu)$ be a DP on Θ and can be represented by $G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}$ and $q : \Theta \to [0, 1]$ be a measurable function. For each k, we independently draw r_k through $P(r_k = 1) = q(\theta_k)$,

$$G_t = \sum_{k:r_k=1} \pi_k^{'} \delta_{\theta_k} \sim DP(q\mu) \tag{2.22}$$

where $\pi_k^{'} = \frac{\pi_k}{\sum_j \pi_j}$ are the re-normalized coefficients for the selected components.

— *Thinned Normalized Random Measures (TNRM).* (Lin et al., 2010; Chen et al., 2012, 2013) Given a NRM $\mu = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}$ on Θ and a bernoulli variable $r_k \in [0, 1]$. The TNRM is,

$$TNRM = \sum_{k:r_k=1} \pi_k^{'} \delta_{\theta_k} \tag{2.23}$$

where $\pi_k^{'} = \frac{\pi_k}{\sum_j \pi_j}$ are the re-normalized coefficients for the selected components. Note that DP is a case of TNRM.

2.3.5 Point-transition

Point-transition is to move the points of a random measure according to an underlying probabilistic transition.

— *Point-transition of PP.* Based on Transition Theorem (Kingman, 1992), let $\Pi \sim PP(\mu)$ be a PP on space Θ and $T : \Theta \times \mathcal{F}_\Theta \to [0, 1]$ be a probabilistic transition. A transformed measure (Lin et al., 2010) is

$$\Pi_p = \{T(\theta) : \theta \in \Pi\} = PP(T\mu) \tag{2.24}$$

— *Point-transition of DP.* (Lin et al., 2010) Let $G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k} \sim DP(\mu)$ and its point-transition is

$$G_p = \sum_{k=1}^{\infty} \pi_k \delta_{T(\theta_k)} \sim DP(T\mu) \tag{2.25}$$

— *Point-transition of Normalized Random Measures (PNRM).* (Chen et al., 2012) Given a NRM $\mu = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}$ on Θ, the point-transition of μ is to draw atoms $\theta_k^{'}$ from a transformed base measure to yield a new NRM as

$$PNRM(\mu) = \sum_k \pi_k \delta_{\theta_k^{'} = T(\theta_k)} \tag{2.26}$$

where T is a transition kernel.

2.3.6 Nesting

With a partition of a space by a process in hand, nesting is to further partition each area by another process. In other words, each component of a random measure is attached an additional random measure.

— *Nested Chinese Restaurant Process (nCRP)* (Blei, Griffiths and Jordan, 2010) A root partition is from a CRP, and each component in this partition is further attached a CRP. Keeping this procedure could form a tree with infinite depth and width.

2.4 Supervised learning

Sometimes data are with different responses/covariants, such as the time tags, geographic locations, GDPs of countries, network relations and so on. How to model the relationship between data and their responses/covariants and applied for future prediction tasks is called supervised learning. According to the method to model above relationship, this thesis summarises the following two categories of researches on supervised BNL.

— *Generalized Linear Models (GLM)-based.* The data are assigned to a number of clusters and the responses (for classification) of data are modeled by multinomial logit models (also called 'softmax') (Shahbaba and Neal, 2009) or GLM (Hannah et al., 2011). DP is used as prior for the weights $\{\pi_k\}_{k=1}^{\infty}$ of clusters and the parameters $\{\theta_k\}_{k=1}^{\infty}$ for the multinomial logit model. Although there is linear relation between data and responses in each cluster, multiple clusters make it possible to capture non-linear relationship. This idea is further extended for group data using HDP (Dai and Storkey, 2015; Zhang et al., 2013).

— *Covariant Space-based.* The relationship between covariant space and data is captured by dependent stochastic processes through: 1) setting the base measure as a special stochastic process, such as single-variable stochastic process (MacEachern, 1999) and multi-variable Gaussian process (Gelfand et al., 2005); 2) varying the stick-breaking procedure, such as linking stick weights by stochastic process (Griffin and Steel, 2006) or kernel function (Dunson and Park, 2008) and permutation stick breaks order (Griffin and Steel, 2006; Chung and Dunson, 2009); 3) auxiliary poisson process on covariant space, such as kernel beta process (Ren et al., 2011) and correlated normalized random measure (Griffin et al., 2013); 4) revised seating mechanisms in CRP and IBP, such as ddCRP (Blei and Frazier, 2011) and ddIBP (Gershman et al., 2015). More discussions on this aspect can be found in the review (Foti and Williamson, 2015).

2.5 Tree structure learning

Tree structure plays an important role in the machine learning due to its pervasive, so learning out a hidden tree from the data also becomes an important branch of BNL. Comparing other efforts on this task, BNL has the ability to learn a tree without the bound of depth and width and to incorporate the new coming data.

— *nCRP-based.* A tree is viewed as a nested sequence of partitions by nCRP. A space is first partitioned by a CRP and each area in this partition is further partitioned into several areas. In this way, a tree with infinite depth and branching could be generated. A datum (i.e., a document) is associated with a path in the tree using DP or Flexible Martingale (Steinhardt and Ghahramani, 2012) prior by nCRP (Blei, Griffiths and Jordan, 2010), and it could associated with a subtree of the generated tree using HDP prior in nHDP (Paisley et al., 2015) instead of a path.

— *Stick-breaking-based.* An iterative stick-breaking process is used to construct a Polya Tree (PT) (R. Daniel Mauldin, 1992) in a nested fashion. A datum is associated with a leaf node of the generated tree; Traditional stick-breaking process is revised to generate breaks with tree structure and results to a Tree Structured Stick Breaking (TSSB) (Ghahramani et al., 2010). A datum is attached to a node in the generated tree.

— *Diffusion-based.* Both Kingman's coalescent (Teh, Daumé and Roy, 2007; Kingman, 1982; Teh et al., 2011) and Dirichlet Diffusion Tree (DDT) (Neal, 2003*a*) define a prior for an infinite (binary) tree. The idea is that the data are generated by a diffusion procedure with several divergences during this procedure. An additional time varying continuous stochastic processes (i.e., Markov process) is needed for the divergence control. A datum is placed at the end of branches of diffusions. DDT is extended to more general structure: multifurcating branches by Pitman Yor Diffusion Tree (PYDT) (Knowles and Ghahramani, 2015) and to feature hierarchy by Beta Diffusion Tree (BDT) (Heaukulani et al., 2014).

2.6 Nonparametric extensions

Considering the power of BNL on the model selection problem, a number of classical learning paradigms with this problem are extended to the nonparametric versions, and these extensions have achieved very good performances on each learning tasks. The idea is to build a Bayesian nonparametric prior through stochastic processes with their manipulations as the model prior and the posterior inference could select the final one. This section names a few nonparametric extensions on the classical models in machine learning area.

— *Gaussian Mixture Model (GMM).* GMM that is a convex combination of a (finite) number of Gaussian distributions is originally proposed as the alternative of single probability distribution for the density estimation task. Comparing a single distribution, GMM could better fit an unknown complex probabilistic density. Infinite GMM (Rasmussen, 1999) as the nonparametric extension of GMM is proposed using DP, where the combining weights and the parameters (i.e., mean and variance) of Gaussian distributions are given a DP prior. In fact, this idea was first developed in as DP mixture model (Ferguson, 1973).

— *Latent Dirichlet Allocation (LDA).* LDA (Blei et al., 2003) is a renowned Bayesian probabilistic graphical model developed for document modeling. The nonparametric extension of LDA is quit straightforward through replacing the Dirichlet and multinomial distributions with DP and multinomial process (Teh, 2006). Another extension LIDA (Archambeau et al., 2015) is to replace Dirichlet distribution by a four-parameter IBP, which allows the number of words to increase with the coming of new documents.

— *Hidden Markov Model (HMM).* HMM (Rabiner, 1989) is a classical and successful model for the time-series data with Markov property. The nonparametric extension of HMM mainly relies on HDP. It is composed by three components: a starting state, a state transition matrix, and an observation matrix. The dimension of sate transition matrix -the number of hidden states-needs to be fixed in advance in traditional HMM. HDP-HMM (Teh, 2006) is the first nonparametric extension by the help of HDP, where each row in state transition matrix is seen as stick weights from DP and the observation from a state is seen as the atoms/parameters from DP. Another three variations are: Sticky HDP-HMM (Fox et al., 2011*b*) for increasing the probability of stating at a state; HDP-HsMM (Johnson and Willsky, 2013) for semi-Markov situations; and iFHMM (Gael et al., 2008) for factorial situations.

— *Nonnegative Matrix/Tensor Factorization (NMF/NTF).* The nonparametric extension of NMF/NTF mainly relies on the machinery of IBP or BP. The initial idea is to factor the data matrix or tensor into a binary (mask) matrix and a factor matrix, and binary (mask) matrix is given a infinite prior, such as IBP (Griffiths and Ghahramani, 2005; Wood and Griffiths, 2006; Ding, Xiang, Molloy, Li et al., 2010) and BP (Liang et al., 2013). Another category of extensions is to assign an infinite prior (i.e., GaP) to the combination parameter of likelihood function.

— *Mixed Membership Stochastic Blockmodels (MMSB).* MMSB (Airoldi et al., 2008) is a famous model for the network/relational data. It defines a (finite) number of blocks for the nodes in the network, and each node could belong to these blocks with different weights (i.e., mixed membership) and assumes that the link between two nodes is determined by the memberships of nodes and the relations between blocks. The (finite) number of blocks are extended to be infinite by HDP (Fan et al., 2015), where the mixed membership for each node is with a sticky HDP-HMM (Fox et al., 2011*b*) fashion.

— *Partially-Observable Reinforcement Learning (PORL).* PORL is for decisions making from incomplete information (i.e., domain knowledge) (Michael and Jordan, 1994). According to the representation methods for the domain knowledge, there are two categories for PORL: hidden variable-based and history-based. For hidden variable-based repression, an efficient tool for this task is the Markov Decision Process (MDP) with finite number of hidden states. A nonparametric extension (Doshi-Velez et al., 2015) for the MDP is based on HDP-HMM, where hidden state transitions are modeled as the stick weights from HDP but the base measure is composed by two parts: one is a function for reward evaluation and the other is for observation generation. For history-based representation, Deterministic Markov Models (DMM) (Mahmud, 2010) are accepted as an efficient tool and PYP (Doshi-Velez et al., 2015) is used as its nonparametric extension with similar fashion with HDP.

2.7 Posterior inference

After building appropriate Bayesian nonparametric models for the specific real-world tasks, the next step is to infer the latent variables (more accurately, the posterior joint distribution of latent variables) in the model with observed data in hand. The most straightforward candidate for the Bayesian model posterior inference is Markov chain Monte Carlo (MCMC) (e.g., Gibbs sampling) which is widely adopted by the Bayesian models including Bayesian nonparametric models. At the same time, there are some other alternatives actively existing in the BNL literature, which are summarized as follow.

2.7.1 Slice sampling

While the (countable) infinite nature enables BNL great modeling power, it also presents a challenge for the model inference: the infinite number of factors and their weights make the posterior inference of the latent variables even harder. One commonly work-around solution in BNL is to use a truncation method. The truncation method (Fox, 2009; Willsky et al., 2009), which uses a relatively big $K^\dagger$ as the (potential) maximum number of topics, is widely accepted but brings an approximation error. Another successful technique to resolve this problem is: *data/variable augmentation* (Van Dyk and Meng, 2001; Tanner and Wong, 2010), also known as *Slice Sampling* (Damlen et al., 1999; Neal, 2003*b*). Next, the existing Bayesian nonparametric models using slice sampling as model inference method are summarised.

— *Slice Sampling for DP.* A realization from DP can be represented as $G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}$, where π_k are (countable infinite number of) stick weights. The slice sampling for DP (Kalli et al., 2011) is to introduce an auxiliary variable $u_i \sim Unif(0, \pi_{k_i})$ which functions as an adaptive truncation for data i, and then only need to sample $\pi > u_i$ for data i.

— *Slice Sampling for BP.* Instead of a fixed range used in DP, a positive decay function $f : \lim_{k\to\infty} f(\pi_k) = 0$ is used to control the support of the auxiliary variable $u_k \sim Unif(0, f(\pi_k))$ (Broderick et al., 2015). Comparing the one used for DP, the decay function is more flexible and applicable to many other models.

2.7.2 Variational inference

Although the methodology of Monte Carlo Markov chain (MCMC) could obtain (in theory) the exact posterior distribution of latent variables, it is not efficient enough. One alternative for BNL is the variational inference, which uses a set of (simpler and independent) variational distributions to approximate the real posterior distribution. It transforms a posterior distribution inference problem to a high-dimensional optimization problem. The work on variational inference for BNL are summarized as follow.

— *Ordinary Variational Inference.* The variational inference for BNL is based on the stick-breaking representation, and the latent variables in this representation include stick weights, atom parameters and data assignment indexes. Since there are possible infinite number of atoms from stochastic processes, such as DP (Blei et al., 2006; Kurihara et al., 2006) and GaP (Roychowdhury and Kulis, 2015), the truncation method is adopted here, and then there are only a finite number of atoms which could be approximated by a finite number of mean-field (latent variable) distributions. Other similar works have been applied for HDP (Blei et al., 2006), IBP (Doshi et al., 2009), and nCRP (Wang and Blei, 2009).

— *Collapsed Variational Inference.* As stated, there are normally three groups of latent variables (i.e., stick weights, atom parameters and data assignment indexes) in BNL. Sometimes stick weights could be marginalized out to make the inferences of DP (Kurihara et al., 2007) and HDP (Teh, Kurihara and Welling, 2007; Sato et al., 2012) more accurate and efficient.

— *Online/Stochastic Variational Inference.* Instead of coordinate-ascent optimization which is not efficient for large dataset because of full pass of the whole data at each iteration, stochastic optimization (Wang et al., 2011; Bryant and Sudderth, 2012) is used for the variational parameter update. At each iteration, a number of data are sampled from the whole dataset and then the variational parameters are updated. This optimization could improve the inference efficiency because of the estimated noisy gradients of the variational objective which are proofed as the natural gradients to the Kullback-Leibler divergence objective. (Hoffman et al., 2013).

— *Truncation-free Variational Inference.* One problem of the above variational inference methods is that the truncation is needed and brings additional approximation error, which is resolved by a locally collapsed method. The global latent variables are marginalized out and the distribution of the local variables could be sampled (Wang and Blei, 2012). This method could be also seen as a combination of sampling and variational inference.

— *Accelerated Variational Inference.* Another idea to avoid truncation is Variational DP (Kurihara et al., 2006), where a tying assumption is given as: the variational distributions of components larger than T are set as prior. T could be adaptively increase as the decreasing of optimization objective. Another kd-tree-based data organization is introduced to accelerate the speed of inference.

2.7.3 Scalability

Considering the exponentially increasing number of data in many areas, the ability of handling this 'big data' also becomes a research direction for BNL. The ides is to extend current inference algorithms particularly designed for single processor/machine into parallel versions for multiple processors/machines. Existing inferences algorithms for this problem are mainly categorized into the following two parts based on their basis (i.e., MCMC and Variational).

— *Parallel MCMC.* A parallel MCMC is proposed for HDP (Smyth et al., 2009) through an asynchronous method. The advantages are easy to incorporate new data and processors and extremely fault-tolerant; the disadvantages is with additional approximation. To overcome this disadvantage, another parallel MCMC for DP, HDP, and PYP (Lovell et al., 2012; Williamson et al., 2013; Dubey et al., 2014) is proposed based on the inverse-superposition of DPs and each processor or machine handel a super-cluster (i.e., one of DPs). Data-based super-clusters generation further improves the efficiency (Chang and Fisher III, 2013). Although above methods could use marginalized representations of DP or HDP to avoid truncation, a slice sampling-based parallel MCMC (Ge et al., 2015) is developed to explicitly sample the stick weights in DP and HDP.

— *Parallel Variational Inference.* The posterior distribution is decomposed for the streaming and parallel inference, and a component identification algorithm is proposed to resolve the mismatch problem when merging different variational posteriors from different processing nodes (Campbell et al., 2015). Instead of obtaining the exact variational approximation of the real posterior distribution by combining subposteriors (from different processing nodes) together, a

Markov chain is designed to obtain the samples from the variational approximation (Neiswanger et al., 2014), which could be seen as a combination of variational and sampling.

2.7.4 Others

Some other inference methods applied for BNL are summarized as follow.

— *Sequential Monte Carlo.* Another inference method is sequential Monte Carlo (also known as particle filtering) which approximates the posterior distribution by a large collection of samples (i.e., particles) which are propagated with time and updated by sequential importance sampling. It has been applied for time varying DP mixture (Caron et al., 2007), Beta-Binomial DP mixture (MacEachern et al., 1999), general conjugate DP mixture (Fearnhead, 2004), nonparametric Bayesian matrix factorization (Wood and Griffiths, 2006).

— *Power Expectation Propagation.* (Minka, 2004) It generalizes expectation propagation and variational inference using a flexible α-divergence. An example is Nonparametric Bayesian Matrix Factorization (Ding, Xiang, Molloy, Li et al., 2010).

2.8 Application scenarios

Although the initial study (Teh, 2006) of BNL in machine learning community is applied for document modeling. As the development of this area, BNL has attracted a lot of attention of researchers with different backgrounds, and therefore many other applications, e.g., image processing and biostatistics, have been considered by BNL. In fact, the increasing number of application scenarios is, in turn, prompting the further development of theories and techniques of this area. A summary of the main fields of application of BNL is described here.

2.8.1 Text mining

The target of text mining is to understand document corpus. According to the difference of target document corpus, applications on text mining are categorized as follows,

— *Single Corpus.* This task aims to learn the shared knowledge across different documents in a corpus. HDP (Teh, 2006) was designed to deal with this problem by generating a global random measure and assigned each document a random measure with global random measure as the base measure. However, the word vocabulary needs to be fixed in advance in HDP, which make it incapable of incorporating new words when new documents are observed. This constraint is further relaxed by Latent IBP Compound Dirichlet Allocation (Archambeau et al., 2015) through a four-parameter IBP.

— *Multiple Corpora.* This task aims to learn the shared knowledge across different corpora. The first attempt was conducted by three-level HDP (Teh, 2006) which generated a global random measure and assigned each corpus a random measure with global random measure as its base measure. Through this hierarchical way, topics shared by different corpora could be inferred. Except the shared topics, the difference between topics at different corpora is also learned by Differential Topic Models (Chen, Buntine, Ding, Xie and Du, 2015) based on PYP.

— *Multiple Time-varying Corpora.* This task aims to learn the shared knowledge across time-varying corpora. Beyond the idea of HDP, an Evolutionary HDP (Zhang et al., 2010) was proposed to deal with this task, which added another time-varying dependency between the random measures for different corpora. The base measure of a corpus is a convex combination of two measures: the global one and the one at the former time stamp.

2.8.2 Natural language processing

The main applications of BNL in the field of natural language processing are detailed below.

— *Word Segmentation.* This task is to identify word boundaries in continuous speech. CRP is used to capture the word sequence generation process (Goldwater et al., 2009) with two options: if a novel lexical item arrives, generating a phonemic form; if not, choosing an existing lexical form. This idea is further applied HDP and achieves better performance than DP.

— *Phrase Alignment.* This task aims to find frequent phrase pairs from bilingual texts which could benefit phrase-based translation systems. Since bilingual texts do not come segmented and the number of aligned phrase pairs is unknown, DP and HDP are adopted as the prior for the emergence of each aligned phrase pair in a probabilistic model for this task (DeNero et al., 2008).

— *Unsupervised Part-of-speech (PoS) Tagging.* Pos tagging is to mark up the words in a text by the corresponding part-of-speeches, which is the basis for the text analysis. Infinite HMM is adopted to model the word sequence by the help HMM and the hidden states (i.e., PoS tags) are unbounded by the help of HDP (Gael et al., 2009).

2.8.3 Computer vision

In the field of computer vision (i.e., image processing and video processing), BNL has been successfully applied to the following problems.

— *Image Interpolation.* This task covers a lot of image processing, mainly including image resize and remap. It tries to factor an image by a linear combination of (finite number of) dictionary atoms (He et al., 2013) which are represented by a matrix. BP and IBP (Zhou et al., 2009) are used as the prior for this matrix to avoid to prefix the dictionary atom number with additional sparsity property. A dependent HBP (Zhou et al., 2011) is proposed to incorporate the patch positions (in an image) constraint.

— *Motion Capture Segmentation.* This task aims to identify the (finite number of) hidden dynamic behaviors in multiple time series and each time series may contain multiple dynamic behaviors. The mapping relationship between time series and dynamic behaviors is expressed by a matrix which are given a BP prior to make it infinite (Fox et al., 2014).

— *Background Subtraction.* This task aims to identify the background from a video stream. Background could be modeled by a probability density which is better to be a multi-mode probability density for the dynamic background. The appropriate number of modes in this distribution depends on the target video stream. DP-based Gaussian Mixture Model (Haines and Xiang, 2014) is proposed to allow mode number determined by the data.

2.8.4 Biology

As the massive amount of data available in Biology, i.e., clinical trials and personalized medicines, BNL provides an efficient paradigm in a fully model-based probabilistic framework with highly flexible and adaptable (Dunson, 2010). This is extremely useful for biology data because of not only the highly multi-dimensional property but also the lack of knowledge or confidence on the parametric model establishing.

— *Brain MRI Tissue Classification.* Since MRI becomes an effective and routine diagnostic tool, the accurate automatic classification of brain MRI could assist the diagnose of doctors. Mixture model clustering algorithms have dominated this task for a long time due to their relatively good performance. DP (da Silva, 2007) and HDP (Jbabdi et al., 2009) are naturally used to extend these algorithms to nonparametric setting and achieved even better performance.

— *Positive Selection Detection.* This task aims to detect the positive natural selection from alignments of protein-coding DNA. Traditional methods assume that the nonsynonymous/synonymous rate ratio at a site is considered as a random variable satisfying a underlying distribution. DP is used as the prior of the nonsynonymous/synonymous rate ratios of site clusters (Huelsenbeck et al., 2006).

— *Expressed Sequence Tag (EST) Analysis.* It is a a fundamental tool for gene identification in organisms. This task aims to estimate how many new genes can be detected in a future EST sample of given size and also to determine the gene discovery rate. Such information is useful for establishing sequencing efficiency in experimental design and for measuring the degree of redundancy of an EST library (Lijoi et al., 2007; Dunson, 2010). PYP (Lijoi et al., 2007) is used to estimate 1) the proportion of genes in the library; 2) the number of new genes; 3) the probability of discovering new genes given additional samples (Favaro et al., 2012).

2.8.5 Music analysis

An interesting application scenario for the BNL is the music analysis which aims for music teaching, analysis of human perception of sounds, and design of music search (Temperley, 2007; Ren et al., 2010). The music data is in the acoustic waveforms. Among a number of methodologies for music analysis, Bayesian techniques have be found very effective and they pave the way for the BNL for the music analysis. The following applications are highlighted.

— *Musical Similarity Computation.* This task is to estimate the timbral similarity between recorded songs, which could be further applied to music retrieval. HDP is used for this task through representing each song by a (stick weight) posterior distribution. The similarity between two songs could be evaluated by the symmetrized KL divergence of two corresponding distributions. Comparing the traditional single Guassian-based or GMM-based approaches, HDP-based approach achieved better performance (Hoffman et al., 2008). Similar idea is also adopted by (Nakano et al., 2012), but a 2-dimensional tree structure is learned to represent each song.

— *Blind Source Separation.* This task is to separate different sources (i.e., instruments and persons) of sound from an audio. One challenge for this task is that the number of sound sources is unknown in advance. A GP-based NMF (Blei, Cook and Hoffman, 2010) is proposed to resolve this problem.

2.9 Summary

The BNL has become a hot topic in machine learning and an efficient tool for the model selection problem. This chapter has reviewed the current state-of-the-art researches in this field: the motivations and definitions, basic stochastic processes and their manipulations, the supervised learning and tree structure learning, nonparametric extensions of classical models, the inference techniques for BNL, and current application scenarios. The reviews have shown that BNL is experiencing its developing stage, so there is still a gap between current theories and techniques with complicated real-world tasks, especially complicated text mining tasks, such as document-word co-clustering and document network learning. Therefore, this study aims to fill this gap by contributing some new theories and techniques to this field, and applies these new theories and techniques on four complicated text mining tasks.

Chapter 3

Bayesian Nonparametric Sparse Nonnegative Matrix Factorization for Document-word Co-clustering

3.1 Introduction

Nonnegative Matrix Factorization (NMF) is a renowned tool for unsupervised learning which has been successfully applied in many research areas (Sandler and Lindenbaum, 2011; He et al., 2011). In text mining area, for example, factorization on the document-word matrix can discover the hidden *topics* from documents, and these topics can assist document clustering for organization or browsing and word clustering for automatic construction of a statistical thesaurus or enhancement of queries (Crouch, 1988). When documents and words are clustered together, it is called co-clustering task (Dhillon, 2001; Bichot, 2010).

Although NMF has experienced a boom, with significant development in recent years, the assumption that the dimensionality of factor matrices need to be predefined is blocking its usage in real-world applications. Therefore, it is more reasonable and practical to automatically discover this number from the data. In the Bayesian nonparametric learning context, IBP (Griffiths and Ghahramani, 2005; Wood and Griffiths, 2006; Ding, Xiang, Molloy, Li et al., 2010) can be seen as the prior of a sparse matrix with an infinite number of columns, which has been adopted for the NMF through $Y = A * X^T = (V \cdot Z) * X^T$ where Z is with a IBP prior. Since only one factor matrix A is with sparsity and nonparametric properties here, it is called Single Nonparametric Sparse NMF (sIBP-NMF) in this chapter. Despite its success on some tasks (such as, factor analysis), sIBP-NMF fails some broader tasks, such as co-clustering shown in the Experiment section, which in fact is the merit of NMF comparing with ordinary factor analysis. Therefore, the idea is to jointly assign two factor matrices IBP priors (named doubly Nonparametric Sparse NMF in this chapter), thus endowing both two factor matrices (A and X) with sparsity and nonparametric properties.

In order to better and more flexibly describe the correlations between corresponding IBP columns than a closely related approach - GP-based dIBP (Williamson et al., 2010), this study instead proposes a framework in which the correlation function is used for the generation of the latent weights pairs $\mu_j^{(1)}$ and $\mu_j^{(2)}$ which are responsible for the generation of $Z_j^{(1)}$ and $Z_j^{(2)}$ respectively. Compared with (Williamson et al., 2010), this work allows both $Z_j^{(1)}$ and $Z_j^{(2)}$ to have much greater flexibility and variations in terms of their non-zero entries. To be specific, this chapter proposes two new dIBPs based on bivariate Beta distribution and copula. Instead of correlating two IBPs at the binary matrices level, the two proposed dIBPs correlate with the two IBPs at the very beginning (at the Beta random variable level). This strategy results in the implementation of the doubly sparse nonparametric NMF framework based on new dIBPs with a simple model structure. Another advantage of the new dIBPs is that the data correlation is directly modeled by the parameters of the bivariate Beta distribution and copula. This correlation is actually a measurement of the focus degree of the hidden topics in the documents, which will be discussed with more details in this chapter. Nevertheless, introducing bivariate Beta distribution and copula presents a challenge for the model inference. This chapter has given three designed inference algorithms for three implementations of the nonparametric NMF framework: GP-based dIBP model, bivariate Beta distribution-based dIBP model, and copula-based dIBP model. The experiments on the synthetic data show the merits of this work comparing the traditional NMF, single IBP-based NMF, and GP-based NMF on the factorization efficiency, sparsity, and correlation flexibility. The experiments on the real-world datasets show that the proposed models perform well on the document word co-clustering task without explicitly predefining the dimension number for the factor matrices, and the models based on new dIBPs have better convergence rates than GP-based dIBP.

The rest of this chapter is organized as follows. Preliminary details of NMF and IBP are briefly introduced in Section 3.2. The dIBP-based doubly sparse nonparametric NMF framework is proposed with three implementations in Section 3.3, and Gibbs samplers are designed for the three models in Section 3.4. Section 3.5 conducts a set of experiments on synthetic data and real-world tasks. Lastly, Section 3.6 summarizes this chapter.

3.2 Preliminary knowledge

3.2.1 Sparse nonnegative matrix factorization

Sparse nonnegative matrix factorization (Ye et al., 2015) is favoured by researchers in several areas due to its output: sparse representation of data. Sparse representation discovers a limited number of components to represent data, which is an important research problem (Heiler and Schnörr, 2006). This sparse representation is generally desirable because it can aid human understanding (e.g., with gene expression data), reduce computational costs, and obtain better generalization in learning algorithms. Sparsity is closely related to feature selection and certain generalizations in machine learning algorithms (Kim and Park, 2008; Yang et al., 2011).

Given a nonnegative matrix $Y_{M\times N}$ (extended to semi-nonnegative by (Ding, Li and Jordan, 2010)), Nonnegative Matrix Factorization (NMF) aims to find two matrices $A_{M\times K}$ and $X_{N\times K}$ to minimize the following cost function

$$J = \|Y_{M\times N} - A_{M\times K}X^T_{N\times K}\|_F^2 + \|A_{M\times K}\|_1 + \|X_{N\times K}\|_1 \tag{3.1}$$

where $\|\cdot\|_F$ is the Frobenius norm, $\|\cdot\|_1$ is the ℓ_1 norm and the elements of A and X are with nonnegative constraint. The ℓ_1 norm in the cost function is used for the sparseness constraint.

NMF is widely used in many application scenarios. Take the document-word co-clustering as an example. The input $Y_{M\times N}$ denotes the frequencies of N words in M documents. The $A_{M\times K}$ denotes the documents' interests on K factors (i.e., topics), and $X_{N\times K}$ denotes the words' interests on K factors (i.e., topics). All documents and words are projected into the same K-dimensional space by NMF. Based on $A_{M\times K}$ and $X_{N\times K}$, document-word co-clustering task can be finished.

One problem of NMF is to determine K. Normally, this variable is experimentally adjusted within a range. Here, this study define the NMF without predefined K as Nonparametric NMF.

3.2.2 Indian buffet process

The Indian Buffet Process (IBP) (Griffiths and Ghahramani, 2005, 2011) is defined as a prior for the binary matrices with an infinite number of columns. The graphical model is shown in Fig. 3.1(a). A stick-breaking construction for IBP (Teh, Görür and Ghahramani, 2007) is proposed as

$$\nu_j \sim Beta(\alpha, 1), \mu_k = \prod_{j=1}^{k} \nu_j, z_{n,k} \sim Bernoulli(\mu_k) \tag{3.2}$$

where $z_{n,k}$ forms a matrix $Z_{N\times K}$, $\{\nu_j\}$ is a set of variables with a Beta distribution, α is the parameter of Beta distribution, and μ_k is the stick weight of column k. The bigger μ_k is, the more 'ones' appear in the column k of the binary matrix $Z_{N\times K}$. The final number of K is determined by the data and the parameter α of IBP.

Some important notations throughout this chapter are summarized in Table 3.1.

3.3 Doubly sparse nonparametric NMF

To set a prior for the infinite matrices A and X in NMF and make sure the two matrices have the same number of columns, this study applies the dependent Indian Buffet processes as the prior for two binary mask matrices of A and X. This section first defines the Doubly Sparse Nonparametric NMF Framework and then introduce its three implementations one by one. Finally, three models are theoretically and empirically compared, and the meaning of the correlation is also discussed.

Table 3.1: Notations in this chapter

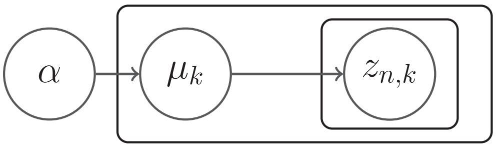
Symbol	meaning in this chapter
M	the row number of Y
N	the column number of Y
K	the latent factor number
$K^{\dagger}$	the truncation level
Y	data matrix with size $M \times N$
$y_{m,n}$	the element of Y at m row and n column
A	factor matrix with size $M \times K$
X	factor matrix with size $N \times K$
$a_{m,k}$	the element of A at m row and k column
$x_{n,k}$	the element of X at n row and k column
$Z^{(1)}$	mask matrix for A with size $M \times K$
$Z^{(2)}$	mask matrix for X with size $N \times K$
$z^{(1)}_{m,k}$	the mask binary variable for element of A at m row and k column
$z^{(2)}_{n,k}$	the mask binary variable for element of X at n row and k column
$V^{(1)}$	loading matrix for A with size $M \times K$
$V^{(2)}$	loading matrix for X with size $N \times K$
$v^{(1)}_{m,k}$	the loading variable for element of A at m row and k column
$v^{(2)}_{n,k}$	the loading variable for element of X at n row and k column
$\mu^{(1)}_k$	k-th stick weight of IBP for A
$\mu^{(2)}_k$	k-th stick weight of IBP for X
θ	model parameters for a bivariate Beta distribution or a copula
$corr$	correlation value

3.3.1 Doubly sparse nonparametric NMF framework

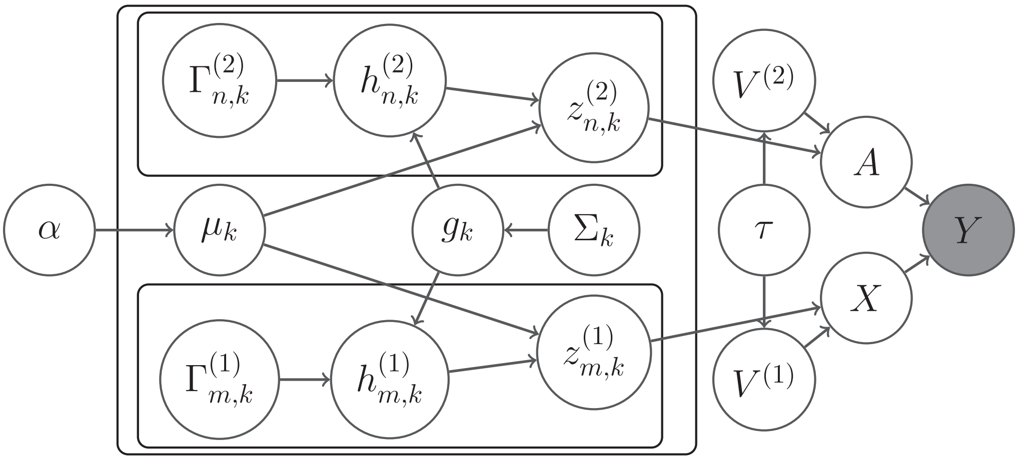
In the Doubly Sparse Nonparametric NMF Framework, the data matrix is modeled as

$$Y = A * X^T = (V^{(1)} \odot Z^{(1)}) * (V^{(2)} \odot Z^{(2)})^T \tag{3.3}$$

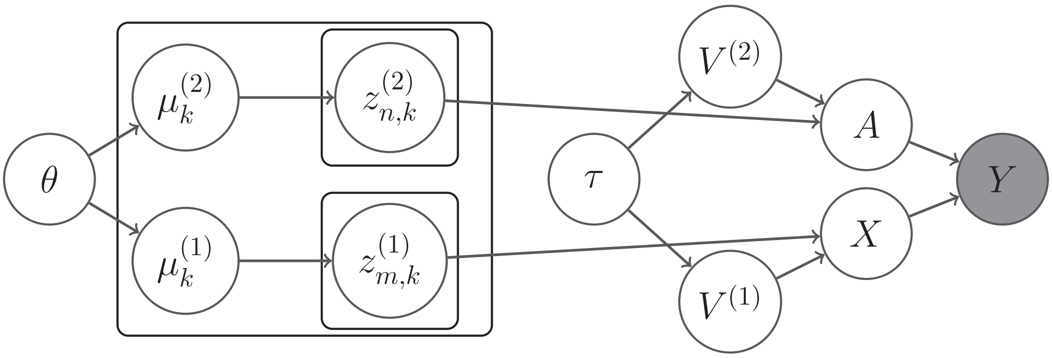
where $Z^{(1)}$ and $Z^{(2)}$ are two binary matrices, $V^{(1)}$ and $V^{(2)}$ are loading matrices, and $\odot$ denotes the Hadamard product. Next, Next, this study assigns these



(a) Indian Buffet Process



(b) Implementation by Gaussian Processes-based dependent Indian Buffet Processes



(c) Implementations by Bivariate-based or Copula-based dependent Indian Buffet Processes

Figure 3.1: Graphical representations for (a) the original Indian Buffet Process, (b) GP-based dependent Indian Buffet Processes and (c) Bivariate Beta distribution-based or Copula-based dependent Indian Buffet Processes.

latent variables probability distributions as

$$
\begin{aligned}
a_{m,k} &= v^{(1)}_{m,k} \cdot z^{(1)}_{m,k}, \;\; v^{(1)}_{m,k} \sim Gam(1, \tau_1) \\
x_{n,k} &= v^{(2)}_{n,k} \cdot z^{(2)}_{n,k}, \;\; v^{(2)}_{n,k} \sim Gam(1, \tau_2)
\end{aligned}
\tag{3.4}
$$

where $Gam(\cdot, \cdot)$ denotes the Gamma distribution and τ_1 and τ_2 are two parameters. It can be seen that $a_{m,k} \geq 0$ and $x_{n,k} \geq 0$ are always satisfied under the above probability distributions. With A and X in hand, the likelihood of the model is defined as

$$y_{m,n}|a_{m,k}, x_{n,k} \sim Exp(y_{m,n}; \sum_k a_{m,k} \cdot x_{n,k} + \epsilon) \tag{3.5}$$

where $Exp(\cdot)$ denotes the Exponential distribution and ϵ is a very small positive number to make the parameter of Exponential distribution greater than zero. The selection of the Exponential distribution is used to guarantee each element of Y with support $[0, +\infty)$. The special parameters for the distributions are designed to retain the desired expectations of these distributions. For example, the expectation of distribution $y_{m,n}$ is $\sum_k a_{m,k} \cdot x_{n,k} + \epsilon$.

Until now, the doubly sparse nonparametric NMF framework construction is finished except for an appropriate prior for $Z^{(1)}$ and $Z^{(2)}$. In the following subsections, three implementations of this framework are introduced using three dIBPs.

3.3.2 Implementation by GP-based dIBP

The first dIBP is proposed based on Gaussian Process (GP) (Williamson et al., 2010). In this GP-based dIBP, each stick weight μ_k is used to generated a different number of columns of binary matrices. For the NMF, this GP-based dIBP is adopted as the prior for the matrices $Z^{(1)}$ and $Z^{(2)}$. The graphical model is shown in Fig. 3.1(b), and the generative process is as

$$\nu_j \sim Beta(\alpha, 1), \ \mu_k = \prod_{j=1}^{k} \nu_j \tag{3.6}$$

where the $\{\mu_k\}$ are IBP sticks as in Eq. (3.2). This set of sticks is shared by two binary matrices through

$$
\begin{aligned}
g_k &\sim GP(0, \Sigma_k) \\
h^{(1)}_{m,k} &\sim GP(g_k, \Gamma^{(1)}_{m,k}) \\
h^{(2)}_{n,k} &\sim GP(g_k, \Gamma^{(2)}_{n,k}) \\
\Sigma_k(t, t') &= \sigma^2 \exp(-\frac{(t-t')^2}{s^2}) \\
\Gamma^{(1)} = \Gamma^{(2)} &= \eta^2 I \\
z^{(1)}_{m,k} &= \delta\{h^{(1)}_{m,k} < F^{-1}(\mu_k | 0, (\Sigma_k)^{(1,1)} + (\Gamma^{(1)}_{m,k})^{(1,1)})\} \\
z^{(2)}_{n,k} &= \delta\{h^{(2)}_{n,k} < F^{-1}(\mu_k | 0, (\Sigma_k)^{(2,2)} + (\Gamma^{(2)}_{n,k})^{(2,2)})\}
\end{aligned}
$$

where $GP(0, \Sigma_k)$ denotes a Gaussian process parameterized by a mean function 0 and a kernel function Σ_k. g_k is a random draw from $GP(0, \Sigma_k)$ which is in turn set as the mean function of $GP(g_k, \Gamma^{(1)}_{m,k})$ and $GP(g_k, \Gamma^{(2)}_{n,k})$ where $\Gamma^{(1)}_{m,k}$ and $\Gamma^{(2)}_{n,k}$ are identify matrices with a parameter η, so there is a hierarchy between different GPs where the correlation is captured. $h^{(1)}_{m,k}$ and $h^{(2)}_{n,k}$ are two random draws from the corresponding GPs, respectively. $F^{-1}(\cdot)$ is the inverse normal cumulative distribution function, and $\delta\{\cdot\}$ is an unit mass function. More details can be found in (Williamson et al., 2010). Since there are only two binary matrices, the GP is degenerated to a two-dimensional Gaussian distribution, and Σ_k is equal to a 2×2 matrix. Note that there is no explicit correlation value between $Z^{(1)}$ and $Z^{(2)}$ in this implementation. Although this model could be used to implement a doubly sparse nonparametric NMF, it nonetheless is inadequate in many matrix factorisation scenarios as the assumption that the total number of non-zero entries are distributed identically does not hold universally. For example, the number of non-zero factors of a Document column may be drastically different to the number of non-zero factors of a corresponding Word column in a document-word matrix.

3.3.3 Implementation by bivariate Beta distribution-based dIBP

There are different ways to link two IBPs. The GP-based dIBP uses the same set of sticks for different binary matrices. Here, this chapter proposes another method that links the initials of two IBPs, ν. In the original IBP, ν satisfies a Beta distribution with parameter $(\alpha, 1)$ as in Eq. (3.2). Therefore, the innovative idea is to find a joint distribution $(\nu^{(1)}, \nu^{(2)})$ with Beta distributions $Beta(\alpha_1, 1)$ and $Beta(\alpha_2, 1)$ as marginal distributions. Following this idea, an intuitive candidate would be Dirichlet distribution. However, there is a strictly negative relation, $\nu^{(1)} + \nu^{(2)} = 1$, between the samples of the Dirichlet distribution, but it hopes to preserve the freedom of two $(\nu^{(1)}, \nu^{(2)})$.

Instead of the Dirichlet distribution, a bivariate Beta distribution (Olkin and Liu, 2003) is adopted here, which probability density function is defined as

$$\begin{aligned} p(r_1, r_2) &= \frac{r_1^{a^0-1} r_2^{b^0-1} (1-r_1)^{b^0+c^0-1} (1-r_2)^{a^0+c^0-1}}{B(a^0, b^0, c^0)(1-r_1 r_2)^{a^0+b^0+c^0}} \\ & s.t., \quad 0 \leq r_1, r_2 \leq 1, \quad a^0, b^0, c^0 > 0 \end{aligned} \tag{3.7}$$

where a^0, b^0, c^0 are three parameters of this distribution and $B(a^0, b^0, c^0)$ is the normalization constant that is hard to be evaluated. One merit of this bivariate Beta distribution is that two marginal distributions are

$$r_1 \sim Beta(a^0, c^0), \quad r_2 \quad \sim Beta(b^0, c^0) \tag{3.8}$$

which satisfies the requirement about Beta marginal distributions of $\nu^{(1)}$ and $\nu^{(2)}$.

Another merit is that this distribution could model positive correlation between (r_1, r_2) with range $[0, 1]$ adjusted by the three parameters (a^0, b^0, c^0) compared to the strictly negative correlation from the Dirichlet distribution. Here, set c^0 of the bivariate Beta distribution to 1 because it must ensure that the marginal distribution of each ν is a Beta distribution with parameter form $(\alpha, 1)$. This condition is for the distribution of the generated binary matrices that satisfy the IBPs. Considering Eq. (3.8), give c^0 a fixed value. Even with a fixed value for c^0, the bivariate Beta distribution in Eq. (3.7) is still able to model different correlations of two variables. In fact, the correlation is a function of a^0 and b^0 when c^0 is fixed

$$corr = \sqrt{a^0 b^0 (a^0+2)(b^0+2)} \cdot \Big[\frac{\Gamma(a+2)\Gamma(b+2)}{(a+b+1)\Gamma(a+b+2)}\mathbb{F} - 1\Big] \tag{3.9}$$

where

$$\mathbb{F} = {}_3F_2(a^0+1, b^0+1, a^0+b^0+1; a^0+b^0+2, a^0+b^0+2; 1) \tag{3.10}$$

is Hypergeometric function[1] that could be evaluated given parameters. The derivative detail of this equation is as follows.

At first, the expectations and variances for r_1 and r_2 of bivariate Beta distribution in Eq. (3.7) with $c^0 = 1$ are

$$\begin{aligned} \mathbb{E}[r_1] &= \frac{a^0}{a^0+1}, \mathbb{V}[r_1] = \frac{a^0}{(a^0+1)^2(a^0+2)} \\ \mathbb{E}[r_2] &= \frac{b^0}{b^0+1}, \mathbb{V}[r_2] = \frac{b^0}{(b^0+1)^2(b^0+2)} \end{aligned} \tag{3.11}$$

[1] https://en.wikipedia.org/wiki/Generalized_hypergeometric_function

and

$$\begin{aligned}\mathbb{E}[r_1 r_2] =& \frac{a^0 b^0 \Gamma(a^0+1)\Gamma(b^0+1)}{(a^0+b^0+1)\Gamma(a^0+b^0+2)} \\ & {}_3F_2(a^0+1, b^0+1, a^0+b^0+1; a^0+b^0+2, a^0+b^0+2; 1)\end{aligned} \tag{3.12}$$

Then,

$$corr(r_1, r_2) = \frac{cov(r_1, r_2)}{\sqrt{\mathbb{V}[r_1]\mathbb{V}[r_2]}} = \frac{\mathbb{E}[r_1 r_2] - \mathbb{E}[r_1]\mathbb{E}[r_2]}{\sqrt{\mathbb{V}[r_1]\mathbb{V}[r_2]}} \tag{3.13}$$

where $corr(r_1, r_2)$ is the correlation between r_1 and r_2; cov is the covariance. After substitution with above equations, Eq. (3.9) is obtained.

For example, when $a^0 = 2.5$ and $b^0 = 4$, the correlation between r_1 and r_2 is 0.995; when $a^0 = 0.05$ and $b^0 = 0.1$, the correlation between r_1 and r_2 is 0.080.

With the desired bivariate Beta distribution in hand, the BiBeta-based dIBP could be built as

$$\begin{aligned}(\nu_k^{(1)}, \nu_k^{(2)}) &\sim biBeta(\theta), \theta : \{a^0, b^0, c^0 = 1\} > 0 \\ z_{n,k}^{(1)} &\overset{\text{i.i.d}}{\sim} Bernoulli(\mu_k^{(1)}), \mu_k^{(1)} = \prod_{j=1}^{k} \nu_j^{(1)} \\ z_{n,k}^{(2)} &\overset{\text{i.i.d}}{\sim} Bernoulli(\mu_k^{(2)}), \mu_k^{(2)} = \prod_{j=1}^{k} \nu_j^{(2)}\end{aligned} \tag{3.14}$$

where $biBeta(\theta)$ denotes the bivariate Beta distribution in Eq. (3.7) and θ denotes the parameters of the distribution. The graphical model is shown in Fig. 3.1(c).

Although the bivariate Beta distribution-based dIBP has extended the freedom of the relation between $\nu^{(1)}$ and $\nu^{(2)}$, the relation is restricted to positive relations in bivariate Beta distribution. Next, this study uses the copula to capture more freedom of relations.

3.3.4 Implementation by copula-based dIBP

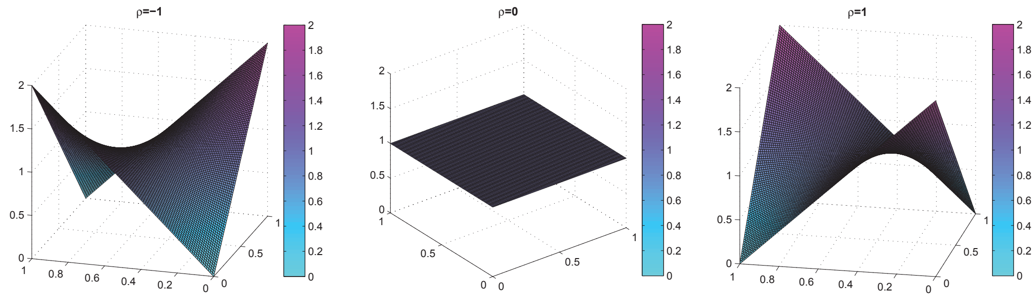


Figure 3.2: FGM copula density surfaces with different values of $\rho = \{-1, 0, 1\}$ and the marginal distributions are both Beta distributions: $Beta(1, 1)$ (means that $\alpha_1 = \alpha_2 = 1$ in Eq. (3.15)).

Copula (Trivedi and Zimmer, 2007) is another alternative for linking two variables with given marginal distributions, and is used to define a joint distribution for variables with known marginal distributions in statistics. Here, this study selects the Farlie-Gumbel-Morgenstern (FGM) Copula (Beare, 2010) as an example.

The definition of the FGM Copula is

$$\begin{aligned} C_\rho(o_1, o_2) &= o_1 o_2 + \rho o_1 o_2 (1 - o_1)(1 - o_2) \\ c_\rho(o_1, o_2) &= 1 + \rho(2o_1 - 1)(2o_2 - 1) \\ \rho &\in [-1, 1] \end{aligned} \tag{3.15}$$

where $C_\rho(o_1, o_2)$ is the joint cumulative distribution function, $c_\rho(o_1, o_2)$ is the joint probability density function, ρ is the parameter of the FGM copula, and o_1 and o_2 are cumulative distribution function values of two marginal distributions that are known in advance. For the problem, two variables are $\nu^{(1)}$ and $\nu^{(2)}$ and their marginal distributions are Beta distributions with parameters $(\alpha_1, 1)$ and

$(\alpha_2, 1)$, so o_1 and o_2 are defined as

$$
\begin{aligned}
o_1 = F^{beta}(\nu^{(1)}) \sim (\nu^{(1)})^{\alpha_1}, \quad f^{beta}(\nu^{(1)}) &\sim \alpha_1 \cdot (\nu^{(1)})^{\alpha_1 - 1} \\
o_2 = F^{beta}(\nu^{(2)}) \sim (\nu^{(2)})^{\alpha_2}, \quad f^{beta}(\nu^{(2)}) &\sim \alpha_2 \cdot (\nu^{(2)})^{\alpha_2 - 1}
\end{aligned}
\tag{3.16}
$$

where $F^{beta}(\nu^{(1)})$ and $f^{beta}(\nu^{(1)})$ represent the cumulative distribution function and probabilistic density function of $\nu^{(1)}$.

In Copula, the correlation of $\nu^{(1)}$ and $\nu^{(2)}$ is modeled or reflected by the value of ρ. As shown in Fig. 3.2, the different correlation (i.e., positive or negative) can be captured by the different value of ρ. The Spearman correlation can be evaluated by $corr = \frac{\rho}{3}$. Since the support of ρ is $[-1, 1]$, the correlation range that can be modelled by the FGM Copula is $[-\frac{1}{3}, \frac{1}{3}]$. In particular, there is no correlation if $\rho = 0$.

Copula-based dIBP is defined by replacing the bivariate Beta distribution in Eq. (3.14) with joint distribution defined by the FGM copula

$$
p(\nu_k^{(1)}, \nu_k^{(2)}) \;=\; c_\rho(\theta), \quad \theta : \{\rho \in [-1, 1], \alpha_1 > 0, \alpha_2 > 0\} \tag{3.17}
$$

The graphical model is the same as the BiBeta-based dIBP in Fig. 3.1(c) but with different parameters θ.

3.3.5 Models discussion

Comparing with GP-based dIBP, one advantage of bivariate Beta distribution-based and copula-based dIBP is that less latent variables are involved. This is easily observed through the graphical models in Fig. 3.1. More latent variables will slow down the convergence of the model inference. Comparing the bivariate Beta distribution, the advantage of Copula is that the cumulative distribution function and probability density function of $(\nu_k^{(1)}, \nu_k^{(2)})$ could be easily obtained,

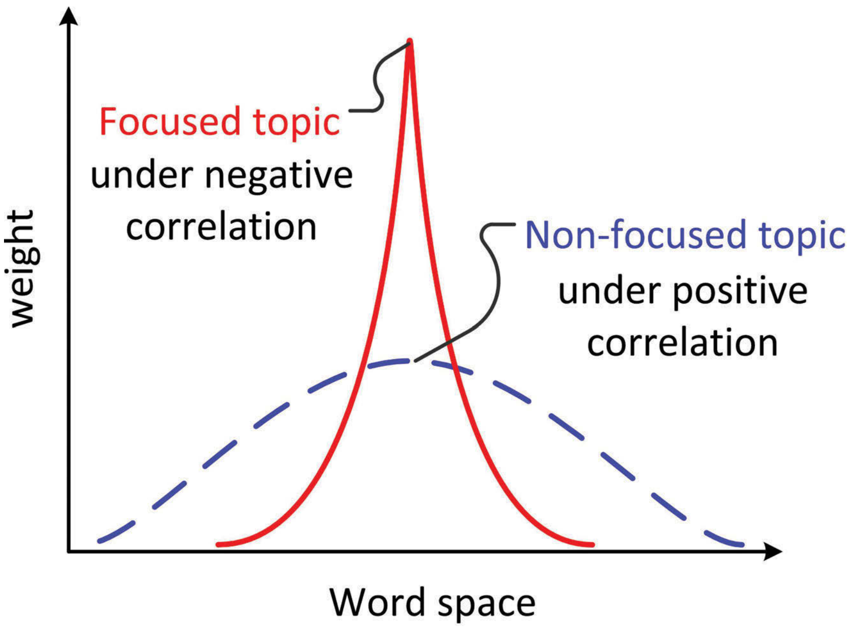


Figure 3.3: Illustration of the meaning of learned correlation. The red/solid curve denotes a focused distribution on words from a large negative correlation; the blue/dashed curve denotes a non-focused distribution on words from a large positive correlation.

but it is only known that the exact form of probability density function of the bivariate Beta distribution. This will impact on the model inference, which will be introduced later.

What does the correlation from bivariate beta distribution-based dIBP NMF and Copula-based dIBP NMF reveal from the data? This study explains its meaning using document-word matrix as an example here. After the factorization, it can obtain two factor matrices $A_{M\times K}$ and $X_{N\times K}$. The K hidden factors are also named *topics* when considering the document datasets. It is known that $\mu_k^{(1)}$ and $\mu_k^{(2)}$ accounts for the generations of $Z_k^{(1)}$ and $Z_k^{(2)}$ that are the k-th column of $Z_{M\times K}^{(1)}$ and $Z_{N\times K}^{(2)}$. If $\mu_k^{(1)}$ is larger, the number of 'ones' in $Z_k^{(1)}$ is greater; the same for $\mu_k^{(2)}$ and $Z_k^{(2)}$. When there is a large negative correlation between $\mu_k^{(1)}$ and $\mu_k^{(2)}$, the number of 'ones' in $Z_k^{(2)}$ is smaller if the number of 'ones' in $Z_k^{(1)}$ is larger. It means that the commonly used topics of documents (the number of 'ones' in $Z_k^{(1)}$ is large) tend to be *focused* topics which only focus on seldom words (the number of 'ones' in $Z_k^{(2)}$ is small), as illustrated by red/solid curve/distribution on words in Fig. 3.3. On the contrary, when there is a large positive correlation between $\mu_k^{(1)}$ and $\mu_k^{(2)}$, the number of 'ones' in $Z_k^{(2)}$ is larger if the number of 'ones' in $Z_k^{(1)}$ is also larger. It means that the commonly used topics of documents (the number of 'ones' in $Z_k^{(1)}$ is large) tend to be *non-focused* topics which are with many words (the number of 'ones' in $Z_k^{(2)}$ is also large), as illustrated by blue/dashed curve/distribution on words in Fig. 3.3. Therefore, it is concluded that the correlation can be seen as the measurement of the focus degree of commonly-used hidden factors/topics in the dataset. On the contrary, the desired topics (i.e., focus or non-focus topics) can be learned from the documents just by fixing this correlation during the algorithms.

3.4 Model inference

With data matrix Y in hand, the objective of this section is to estimate the hidden variables by a properly designed MCMC (Gibbs) sampler for their posterior distribution, $p(\mu, Z, V, \theta|Y)$. It is difficult to perform posterior inference under infinite mixtures, thus a common work-around solution in Bayesian nonparametric learning is to use a truncation method. The truncation method, which uses a relatively big $K^\dagger$ as the (potential) maximum number of factors, is widely accepted.

3.4.1 Inference for the GP-based dIBP

It needs to highlight that GP-based dIBP is not the contribution of this study. The contribution here is to link the GP-based dIBP with the nonnegative matrix factorization likelihood. The inference of this model is given here for the self-contained purpose. The details can be found in (Williamson et al., 2010). The conditional distributions are

Sampling μ

$$p(\mu_k|\cdots) \propto \frac{\mu_K^\alpha}{\mu_k} \prod_{t=1}^{2} \prod_{n}^{N_t} (\gamma_k^t)^{z_{n,k}^t} (1-\gamma_k^t)^{1-z_{n,k}^t} \tag{3.18}$$

where

$$\gamma_k^t = F(F^{-1}(\mu_k|0, \Sigma_k^{(t,t)} + \eta^2) - g_k^t|0, \eta^2) \tag{3.19}$$

where $F()$ is normal cumulative distribution function.

Sampling g

$$p(g_k|\cdots) \propto \mathcal{N}(g_k|0, \Sigma_k) \cdot \prod_{t} \prod_{n}^{N_t} \mathcal{N}(h_{n,k}^t|g_k^t, \eta^2) \tag{3.20}$$

where $\mathcal{N}()$ denotes normal distribution.

Sampling h

$$p(h_{n,k}^t|\cdots) \propto \mathcal{N}(g_k^t, \eta^2) \tag{3.21}$$

with support

$$\begin{cases} h_{n,k}^t \in (-\infty, \widetilde{\mu}_k^t] & \text{if } z_{n,k}^t = 1 \\ h_{n,k}^t \in [\widetilde{\mu}_k^t, +\infty) & \text{if } z_{n,k}^t = 0 \end{cases} \tag{3.22}$$

where

$$\widetilde{\mu}_k^t = F^{-1}(\mu_k|0, \Sigma_k^{(t,t)} + \eta^2) \tag{3.23}$$

Sampling s

$$P(s|\cdots) \propto Gam(s; hs, 1) \prod_{k}^{K} \mathcal{N}(g_k|0, \Sigma_k) \tag{3.24}$$

Sampling Z

$$
\begin{aligned}
p(z_{m,k}^{(1)} = 1|\cdots) &\propto \gamma_k^1 \prod_n Exp(y_{m,n}; \sum_l v_{m,l}^{(1)} z_{m,l}^{(1)} v_{n,l}^{(2)} z_{n,l}^{(2)} + \epsilon) \\
p(z_{m,k}^{(1)} = 0|\cdots) &\propto (1-\gamma_k^1) \prod_n Exp(y_{m,n}; \sum_l v_{m,l}^{(1)} z_{m,l}^{(1)} v_{n,l}^{(2)} z_{n,l}^{(2)} + \epsilon)
\end{aligned} \tag{3.25}
$$

and

$$
\begin{aligned}
p(z_{n,k}^{(2)} = 1|\cdots) &\propto \gamma_k^2 \prod_m Exp(y_{m,n}; \sum_l v_{m,l}^{(1)} z_{m,l}^{(1)} v_{n,l}^{(2)} z_{n,l}^{(2)} + \epsilon) \\
p(z_{n,k}^{(2)} = 0|\cdots) &\propto (1-\gamma_k^2) \prod_m Exp(y_{m,n}; \sum_l v_{m,l}^{(1)} z_{m,l}^{(1)} v_{n,l}^{(2)} z_{n,l}^{(2)} + \epsilon)
\end{aligned} \tag{3.26}
$$

where γ_k^t is same as in Eq. (3.19).

3.4.2 Update stick weights

When updating μ_k, it needs to find its conditional distribution given μ_{k-1} and μ_{k+1}, because the order of μ must be maintained to make the marginal distributions of Z satisfy two IBPs. Based on M-H sampler, the acceptance ratio of a new sample for $\mu_k = [\mu_k^{(1)}, \mu_k^{(2)}]$ is

$$
\min\left(1, \frac{p(Z|\mu_k^*) \cdot p(\mu_k^*|\mu_{k+1}, \mu_{k-1})}{p(Z|\mu_k) \cdot p(\mu_k|\mu_{k+1}, \mu_{k-1})} \times \frac{q(\mu_k)}{q(\mu_k^*)}\right) \tag{3.27}
$$

where $q(\cdot)$ is a proposal distribution which will be explained later; μ_k^* is a new sample drawn from proposal distribution $q(\cdot)$; $p(Z|\mu_k)$ is the likelihood of μ_k to generate kth column of binary matrix Z as in Eq. (3.14) and the $p(\mu_k^*|\mu_{k+1}, \mu_{k-1})$ is the conditional probability density function of μ_k^* within the range of $[\mu_{k+1}, \mu_{k-1}]$, which will be different when different strategies in Section 3.3 are used to link $\mu_k^{(1)}$ and $\mu_k^{(2)}$.

Next, derive the conditional probability density function of $p(\mu_k|\mu_{k+1}, \mu_{k-1})$

as follows: For the first column,

$$\mu_1^{(1)} = \nu_1^{(1)}, \quad \mu_1^{(2)} = \nu_1^{(2)} \tag{3.28}$$

and

$$p(\mu_1^{(1)}, \mu_1^{(2)}) = p(\nu_1^{(1)}, \nu_1^{(2)}) \tag{3.29}$$

where $p(\cdot, \cdot)$ is the joint probability density function given by Eq. (3.7) or Eq. (3.15) from two proposed dIBPs. For the second column,

$$\mu_2^{(1)} = \nu_2^{(1)}\mu_1^{(1)}, \quad \mu_2^{(2)} = \nu_2^{(2)}\mu_1^{(2)} \tag{3.30}$$

and

$$\begin{aligned}
p(\mu_2^{(1)}, \mu_2^{(2)}) &= p(\nu_2^{(1)} \cdot \mu_1^{(1)}, \nu_2^{(2)} \cdot \mu_1^{(2)}) \\
&= \frac{p(\nu_2^{(1)}, \nu_2^{(2)})}{|J_1|} \\
&= \frac{p\left(\mu_2^{(1)}/\mu_1^{(1)}, \mu_2^{(2)}/\mu_1^{(2)}\right)}{\mu_1^{(1)} \cdot \mu_1^{(2)}}
\end{aligned}$$

where J_1 is the Jacobian matrix as

$$J_1 = \begin{bmatrix} \mu_1^{(1)} & 0 \\ 0 & \mu_1^{(2)} \end{bmatrix} \tag{3.31}$$

For the k-th column,

$$\mu_k^{(1)} = \nu_k^{(1)}\mu_{k-1}^{(1)}, \quad \mu_k^{(2)} = \nu_k^{(2)}\mu_{k-1}^{(2)} \tag{3.32}$$

and

$$\begin{aligned} p(\mu_k^{(1)}, \mu_k^{(2)}) &= p(\nu_k^{(1)} \cdot \mu_{k-1}^{(1)}, \nu_k^{(2)} \cdot \mu_{k-1}^{(2)}) \\ &= \frac{p(\nu_k^{(1)}, \nu_k^{(2)})}{|J_{k-1}|} \\ &= \frac{p\left(\mu_k^{(1)}/\mu_{k-1}^{(1)}, \mu_k^{(2)}/\mu_{k-1}^{(2)}\right)}{\mu_{k-1}^{(1)} \cdot \mu_{k-1}^{(2)}} \end{aligned}$$

where J_{k-1} is the Jacobian matrix as

$$J_{k-1} = \begin{bmatrix} \mu_{k-1}^{(1)} & 0 \\ 0 & \mu_{k-1}^{(2)} \end{bmatrix} \tag{3.33}$$

To summarize, the conditional density of μ_k is

$$p(\mu_k|\mu_{k-1}, \mu_{k+1}) \propto \frac{p\left(\mu_k^{(1)}/\mu_{k-1}^{(1)}, \mu_k^{(2)}/\mu_{k-1}^{(2)}\right)}{\mu_{k-1}^{(1)} \cdot \mu_{k-1}^{(2)}} \cdot \frac{p\left(\mu_{k+1}^{(1)}/\mu_k^{(1)}, \mu_{k+1}^{(2)}/\mu_k^{(2)}\right)}{\mu_k^{(1)} \cdot \mu_k^{(2)}} \tag{3.34}$$

Considering the support of μ_k, the proposal distribution, $q(\cdot)$, is set as the product of two independent truncated Beta distributions

$$\begin{aligned} q(\mu_k) &= Beta\left(\mu_k^{(1)}; \frac{\alpha_1}{K^\dagger}, 1\right) \cdot Beta\left(\mu_k^{(2)}; \frac{\alpha_2}{K^\dagger}, 1\right) \\ \mu_k^{(1)} &\sim Beta\left(\frac{\alpha_1}{K^\dagger}, 1\right),\ \mu_k^{(1)} \in [\mu_{k+1}^{(1)}, \mu_{k-1}^{(1)}] \\ \mu_k^{(2)} &\sim Beta\left(\frac{\alpha_2}{K^\dagger}, 1\right),\ \mu_k^{(2)} \in [\mu_{k+1}^{(2)}, \mu_{k-1}^{(2)}] \end{aligned}$$

It is quite easy to sample the truncated Beta distribution because Beta distribution is a standard distribution.

3.4.3 Update binary matrices

Two binary matrices, $Z : \{Z^{(1)}, Z^{(2)}\}$, can be updated separately. Each element in two matrices satisfies a Bernoulli distribution with the following conditional probabilities

$$\begin{aligned} p(z^{(1)}_{m,k} = 1) &\propto \mu^{(1)}_k \prod_n Exp(y_{m,n}; \sum_l a_{m,l} \cdot x_{n,l} + \epsilon) \\ &\propto \mu^{(1)}_k \prod_n Exp(y_{m,n}; \sum_l v^{(1)}_{m,l} z^{(1)}_{m,l} v^{(2)}_{n,l} z^{(2)}_{n,l} + \epsilon) \end{aligned} \tag{3.35}$$

and

$$\begin{aligned} p(z^{(1)}_{m,k} = 0) &\propto (1 - \mu^{(1)}_k) \prod_n Exp(y_{m,n}; \sum_l a_{m,l} \cdot x_{n,l} + \epsilon) \\ &\propto (1 - \mu^{(1)}_k) \prod_n Exp(y_{m,n}; \sum_l v^{(1)}_{m,l} z^{(1)}_{m,l} v^{(2)}_{n,l} z^{(2)}_{n,l} + \epsilon) \end{aligned} \tag{3.36}$$

and

$$\begin{aligned} p(z^{(2)}_{n,k} = 1) &\propto \mu^{(2)}_k \prod_m Exp(y_{m,n}; \sum_l a_{m,l} \cdot x_{n,l} + \epsilon) \\ &\propto \mu^{(2)}_k \prod_m Exp(y_{m,n}; \sum_l v^{(1)}_{m,l} z^{(1)}_{m,l} v^{(2)}_{n,l} z^{(2)}_{n,l} + \epsilon) \end{aligned} \tag{3.37}$$

and

$$\begin{aligned} p(z^{(2)}_{n,k} = 0) &\propto (1 - \mu^{(2)}_k) \prod_m Exp(y_{m,n}; \sum_l a_{m,l} \cdot x_{n,l} + \epsilon) \\ &\propto (1 - \mu^{(2)}_k) \prod_m Exp(y_{m,n}; \sum_l v^{(1)}_{m,l} z^{(1)}_{m,l} v^{(2)}_{n,l} z^{(2)}_{n,l} + \epsilon) \end{aligned} \tag{3.38}$$

where ϵ is small positive number. Since elements of Z are with discrete distribution, it is easy to obtain samples.

3.4.4 Update loading matrices

Since the prior for V is Gamma distribution and the likelihood is exponential distribution, the conditional distribution for V is

$$\begin{aligned} p(v^{(1)}_{m,k}|z^{(1)}_{m,k}, a_{m,k}) &\propto Gam(v^{(1)}_{m,k}; \alpha_1) \prod_n Exp(y_{m,n}; \sum_k a_{m,k} \cdot x_{n,k} + \epsilon) \\ &\propto Gam(v^{(1)}_{m,k}; \alpha_1) \prod_n Exp(y_{m,n}; \sum_k v^{(1)}_{m,k} z^{(1)}_{m,k} v^{(2)}_{n,k} z^{(2)}_{n,k} + \epsilon) \end{aligned} \tag{3.39}$$

and

$$\begin{aligned} p(v^{(2)}_{n,k}|z^{(2)}_{n,k}, x_{n,k}) &\propto Gam(v^{(2)}_{n,k}; \alpha_2) \prod_m Exp(y_{m,n}; \sum_l a_{m,l} \cdot x_{n,l} + \epsilon) \\ &\propto Gam(v^{(2)}_{n,k}; \alpha_2) \prod_m Exp(y_{m,n}; \sum_l v^{(1)}_{m,l} z^{(1)}_{m,l} v^{(2)}_{n,l} z^{(2)}_{n,l} + \epsilon) \end{aligned} \tag{3.40}$$

Since the existence of the ϵ, the posterior of V is not a Gamma distribution, so M-H sampler could be adopted to get its samples.

3.4.5 Update model parameter

In Fig. 3.1(c), the graphical model has parameter θ. For the different strategies to link $(\nu^{(1)}, \nu^{(2)})$, the parameters must be different. Therefore, it needs to design corresponding update methods for the proposed two strategies: bivariate Beta distribution and copula.

Bivariate Beta Distribution

The parameters of bivariate Beta distribution, $\theta : \{a^0, b^0\}$, are given two Gamma priors. The conditional distributions are

$$p([a^0\ b^0]|\cdots) \propto Gam([a^0\ b^0]; hp) \prod_{k=1}^{K} p(\mu_k^{(1)}, \mu_k^{(2)}|a^0, b^0) \tag{3.41}$$

where hp is the hyper-parameter of the prior for a^0 and b^0 and K is the number of active columns of Z. The 'active' column means that there is at least one element with 1 in that column.

Copula

There are three parameters for each copula, $\theta : \{\rho, \alpha_1, \alpha_2\}$. Their conditional distributions are

$$p([\alpha_1\ \alpha_2]|\cdots) \propto Gam([\alpha_1\ \alpha_2]; hp) \prod_{k=1}^{K} c(\mu_k^{(1)}, \mu_k^{(2)}) \tag{3.42}$$

where $c(\cdot,\cdot)$ is the copula density in Eq. (3.15). ρ of the FGM copula is given a uniform distribution on its support $[-1, 1]$, and its posterior is

$$p(\rho|\cdots) \propto \prod_{k=1}^{K} c(\mu_k^{(1)}, \mu_k^{(2)}|\rho) \tag{3.43}$$

After introducing update methods for all the latent variables, the inference (i.e., Gibbs sampler) for the three models are summarized in Algorithm 3.1 for the GP-based dIBP NMF (GP-dIBP-NMF) model, Algorithm 3.2 for the bivariate Beta distribution-based dIBP NMF (BB-dIBP-NMF) model, and Algorithm 3.3 for the Copula-based dIBP NMF (C-dIBP-NMF) model.

Algorithm 3.1: Gibbs Sampler for GP-dIBP-NMF

Input: Y
Output: A, X
initialization;
while $i \leq max_{iter}$ **do**
  // latent variables of dIBP
  Update μ by Eq. (3.18);
  Update Z by Eq. (3.25) and (3.26);
  Update g by Eq. (3.20);
  Update h by Eq. (3.21);
  Update s by Eq. (3.24);
  // latent variables of NMF
  Update V by Eq. (3.39) and (3.40);
  $i++$;
return A and X;

3.5 Experiments

In this section, a series experiments on the synthetic data is conducted to show the merits of this work comparing the traditional NMF, single IBP-based NMF, and GP-based NMF on the the sparsity and nonparametric properties (Section 3.5.1), and correlation flexibility (Section 3.5.2). A real-world task: document-word co-clustering is conducted to show the usefulness of this work comparing other models (Section 3.5.3).

3.5.1 Evaluation on sparsity and nonparametric properties

Randomly generate a matrix $Y_{20\times30}$ using the following procedure: 1) give a vector $[0.5, 0.4, 0.3, 0.2, 0.1]$ as the parameters of five Bernoulli distributions; 2) randomly generate the elements of i-th column (i.e., $A(:, i)$ and $X(:, i)$) of both two matrices $A_{20\times5}$ and $X_{30\times5}$ using the Bernoulli distribution with i-th value of above vector; 3) generate Y as the product of A and X as $Y = AX^T$. Since the values of parameters of Bernoulli distributions are small, the generated factor

---

Algorithm 3.2: Gibbs Sampler for BB-dIBP-NMF

Input: Y
Output: A, X
initialization;
while $i \leq max_{iter}$ **do**
  // latent variables of dIBP
  Update μ by Eq. (3.27);
  Update Z by Eq. (3.35), (3.36), (3.37) and (3.38);
  Update a^0 and b^0 by Eq. (3.41);
  // latent variables of NMF
  Update V by Eq. (3.39) and (3.40);
  $i++$;
return A and X;

---

matrices A and X tend to be sparse. Here, the matrix Y is used as the input data for different algorithms (i.e., traditional sparse NMF (sNMF) in Eq. (3.1), single nonparametric sparse NMF (sIBP-NMF), and the proposed dIBP-NMF), and then the sparsity of learned factor matrices from different algorithms are evaluated and compared. In order to quantitatively compare the sparsity from different algorithms, the following metrics are designed

$$\begin{aligned} S_A &= \sum_m \sum_k \mathbb{1}(A(m,k) = 0) \\ S_X &= \sum_n \sum_k \mathbb{1}(X(n,k) = 0) \end{aligned} \tag{3.44}$$

where $\mathbb{1}(\cdot)$ is an indicator function parameterized by a condition and equals to 1 if condition is satisfied; 0, otherwise. Here, the sparsity of A and X is evaluated separately considering the sIBP-NMF. Note that the 0.00000001 acts as a relax of the sparsity in the implementation. In the experiments, randomly generate a number 100 of $Y_{20\times 30}$ using above procedure. The results are shown in Fig. 3.4, which has shown and compared the sparsity of learned factor matrices A and X from different algorithms. Note that the results are average of 100

Algorithm 3.3: Gibbs Sampler for C-dIBP-NMF

Input: Y
Output: A, X
initialization;
while $i \leq max_{iter}$ **do**
    // *latent variables of dIBP*
    Update μ by Eq. (3.27);
    Update Z by Eq. (3.35), (3.36), (3.37) and (3.38);
    Update α_1 and α_2 by Eq. (3.42);
    Update ρ by Eq. (3.43);
    // *latent variables of NMF*
    Update V by Eq. (3.39) and (3.40);
    $i++$;
return A and X;

trials. The x-axis denotes the number of factors for NMF and sNMF since they need it as an input but sIBP-NMF and dIBP-NMF do not. It can be seen that the sNMF obtains more sparsity comparing NMF by the help of the sparse constraint. Apparently, the sparsity is increasing as the number of factors increasing too. Since sIBP-NMF and dIBP-NMF do not need the factor number as an input, the values of S_A and S_X from two algorithms are equal in the figure. For each trial, a distribution on factor number can be obtained as two examples shown in Fig. 3.5 for sIBP-NMF and dIBP-NMF with 5,000 Gibbs samples. The peak of the distribution is seen as the final learned factor number from the algorithm. The statistics on learned factor numbers from 100 trails are shown in the right subfigure in Fig. 3.5. The averages are around the real one (i.e., 5 in the generative procedure), which denotes the relatively accurate on the factor learning (i.e., nonparametric property). Not surprisingly, sIBP-NMF has good sparsity of A but not the X (the value of S_A is large even comparing the one from sNMF but S_X is small). Its sparsity on A is due to its IBP prior. Since there are two IBP priors for A and X in dIBP-NMF, the resultant A and X are both sparse. Therefore, a conclusion could be drawn that the proposed

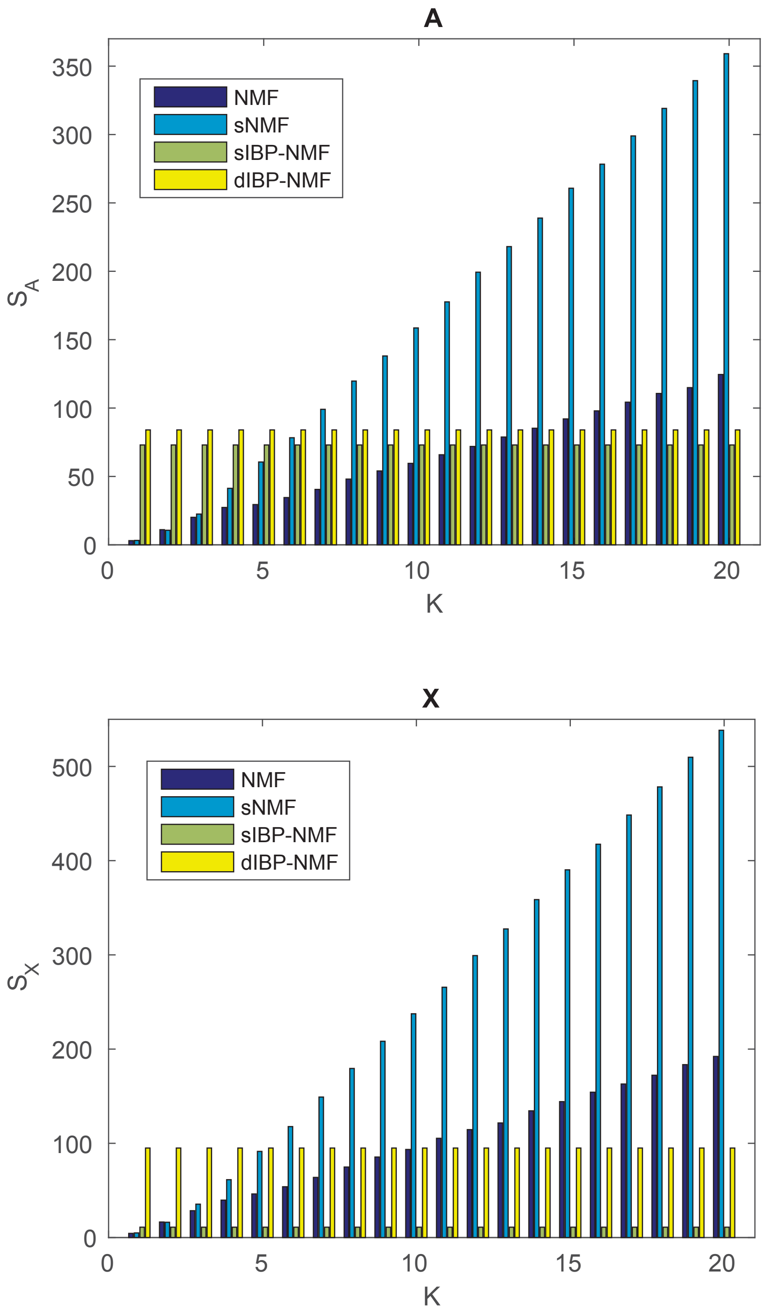


Figure 3.4: Comparison of the sparsity on synthetic dataset between traditional NMF (NMF), traditional sparse NMF (sNMF), single IBP-based sparse NMF (sIBP-NMF), and doubly IBP-based sparse NMF (dIBP-NMF). Note that dIBP-NMF here is based on bivariate beta distribution.

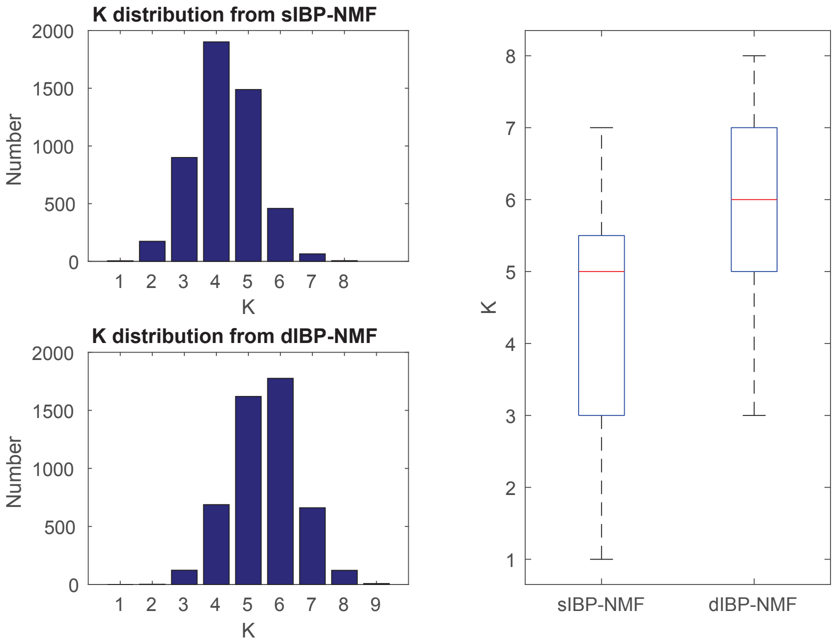


Figure 3.5: Comparison of the learned topic number distribution on synthetic dataset between single IBP-based sparse NMF (sIBP-NMF) and doubly IBP-based sparse NMF (dIBP-NMF). Note that dIBP-NMF here is based on bivariate beta distribution.

dIBP-NMF could obtain two sparse factor matrices but sIBP-NMF could only obtain one, which will impact on the ability to do co-clustering task that will be demonstrated in Section 3.5.4.

3.5.2 Evaluation on correlation flexibility

As claimed, the proposed model is with more flexibility to allow the numbers of non-zero entries of factor matrices more different from each other comparing GP-based dIBP. In order to show this flexibility, a metric needs to be designed firstly to measure this flexibility by comparing the number of non-zero entries of two factor matrices (A and X) from the models: the mean of the differences

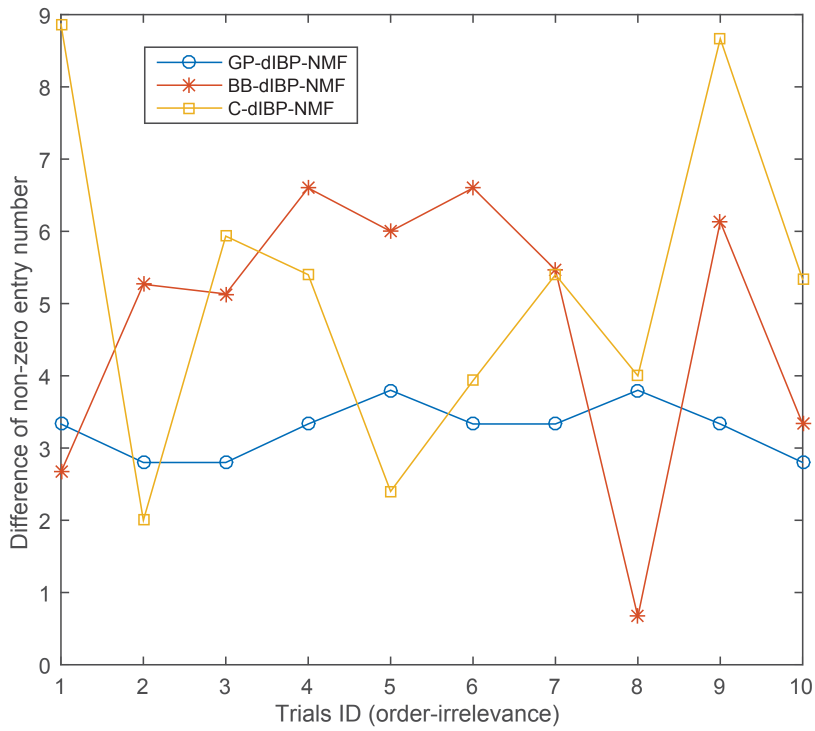


Figure 3.6: Results on synthetic data to show the flexibility of the different models. The x-axis denotes the trial IDs (order is irrelevant).

between the corresponding columns of A and X as

$$\frac{1}{K}\sum_{k=1}^{K}|N_k^{(1)} - N_k^{(2)}|$$

where K is the number of columns of both matrices, $N_k^{(1)}$ is the number of non-zero entries of k-th column of A, and $N_k^{(2)}$ is the number of non-zero entries of k-th column of X. Apparently, the larger this metric is, the more flexibility a model has. There are ten randomly generated matrices with same size: 20×30, and run three models on ten matrices. The designed metric has been evaluated by on the learned factor matrices of different models. As shown in Fig. 3.6, it can

Table 3.2: Evaluation metrics for clustering

Jaccard Coefficient	$JC = \frac{a}{a+b+c}$
Folkes & Mallows	$FM = \left(\frac{a}{a+b} \cdot \frac{a}{a+c}\right)^{1/2}$
F1 measure	$F1 = \frac{2a^2}{2a^2+ac+ab}$

be seen that the metrics on BB-dIBP-NMF and C-dIBP-NMF are larger than GP-dIBP-NMF in most trails and are also with more fluctuations and larger non-zero entry number differences comparing with GP-dIBP-NMF. It concludes that the proposed dIBPs are more flexible than GP-based dIBP.

3.5.3 Real-world task: document-word co-clustering

In this subsection, the proposed algorithms is applied to a real-world task: document-word co-clustering. The real-world datasets[2] used for this task are:

- **Cora Dataset** The Cora dataset consists of 2708 scientific publications classified into one of seven classes. The dictionary consists of 1433 unique words.

- **Citeseer Dataset** The CiteSeer dataset consists of 3312 scientific publications. The dictionary consists of 3703 unique words. The labels of these papers are set as their research areas, such as AI (Artificial Intelligence), ML (Machine Learning), Agents, DB (Database), IR (Information Retrieval) and HCI (Human-Computer Interaction).

The above datasets are already with benchmarks for the document clusters but without benchmarks for the word clusters. The co-occurrence relations are used between words to generate a distance matrix through which a number (10, here)

[2]http://linqs.cs.umd.edu/projects/projects/lbc/

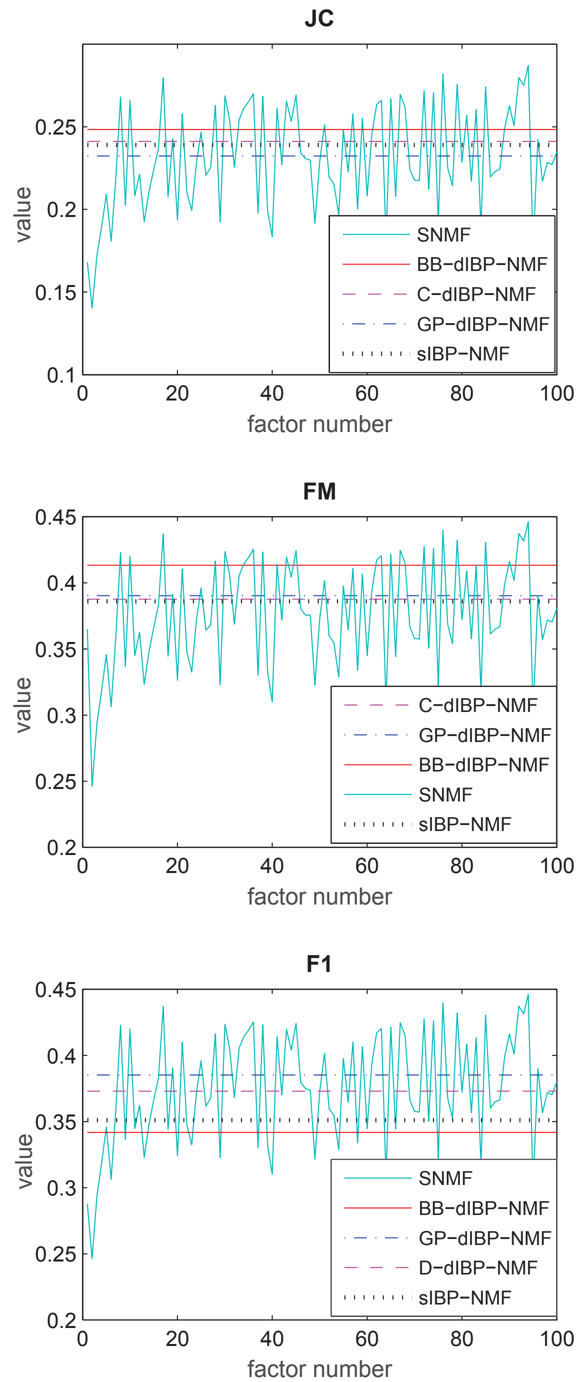


Figure 3.7: Document-word co-clustering (document side) comparisons on *Citeseer* dataset between GP-dIBP-NMF, BB-dIBP-NMF, C-dIBP-NMF, sIBP-NMF, and SNMF.

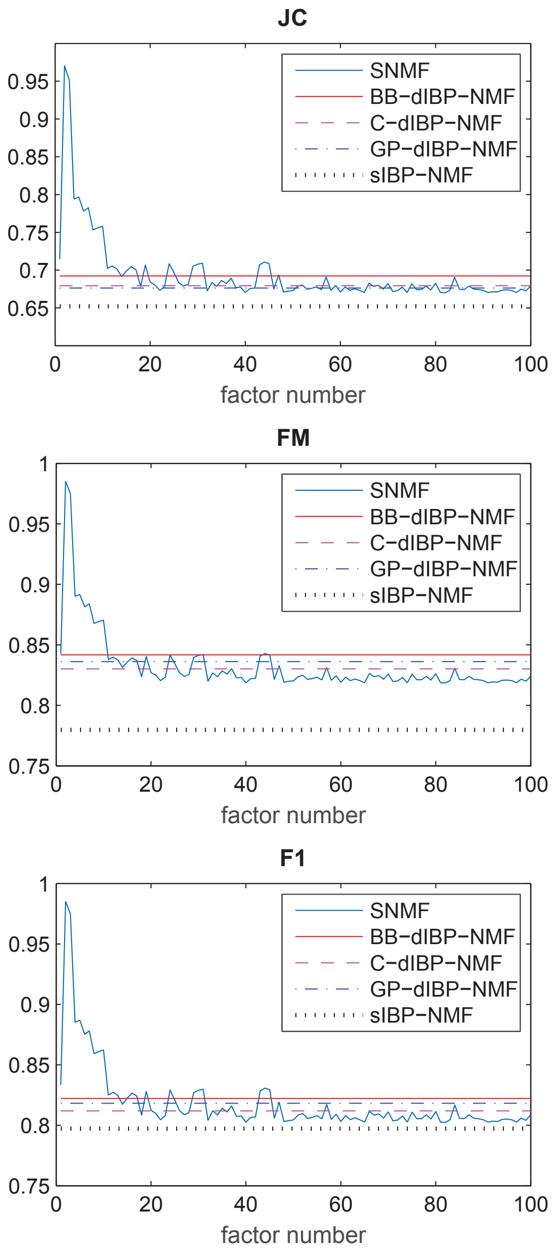


Figure 3.8: Document-word co-clustering (word side) comparisons on *Citeseer* dataset between GP-dIBP-NMF, BB-dIBP-NMF, C-dIBP-NMF, sIBP-NMF, and SNMF.

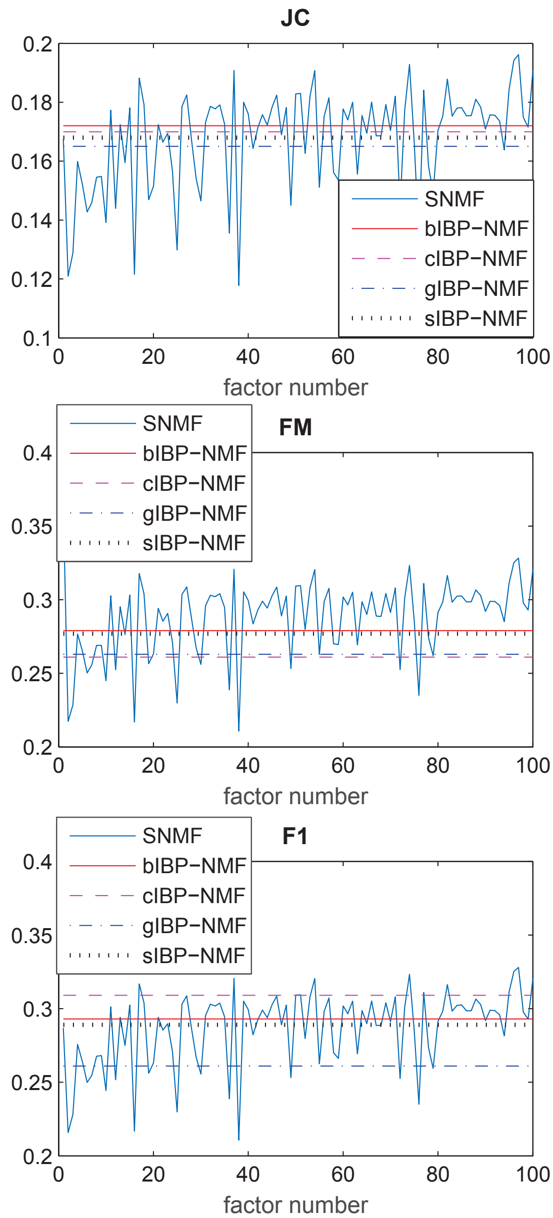


Figure 3.9: Document-word co-clustering (document side) comparisons on *Cora* dataset between GP-dIBP-NMF, BB-dIBP-NMF, C-dIBP-NMF, sIBP-NMF, and SNMF.

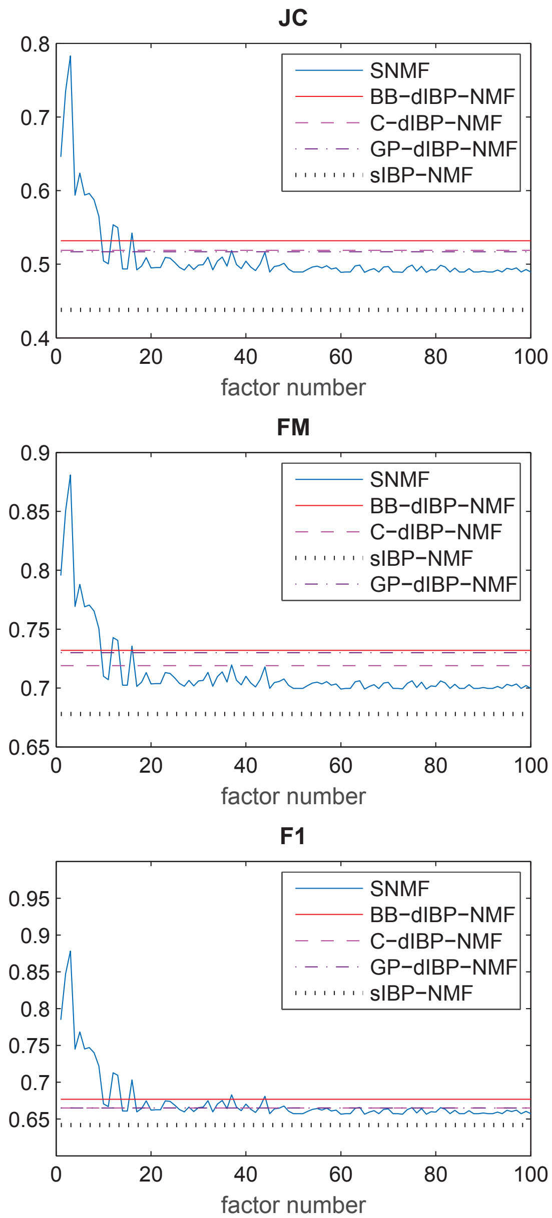


Figure 3.10: Document-word co-clustering (word side) comparisons on *Cora* dataset between GP-dIBP-NMF, BB-dIBP-NMF, C-dIBP-NMF, sIBP-NMF, and SNMF.

of word clusters can be obtained by spectral clustering algorithm. These word clusters are seen as benchmarks. The evaluation metrics for clustering are Jaccard Coefficient (JC), Folkes&Mallows (FM) and F1 measure (F1). Given a clustering result,

- a is the number of two points that are in the same cluster of both benchmark results and clustering results;
- b is the number of two points that are in the same cluster of benchmark results but in different clusters of clustering results;
- c is the number of two points that are not in the same cluster of the two benchmark results but are in the same cluster of clustering results.

and three metrics are computed the equations in Table 3.2 (bigger means better).

The comparisons with other models on the document-word co-clustering task are conducted, including sparse NMF (SNMF) in Eq. (3.1), single IBP-based sparse NMF (sIBP-NMF) (Griffiths and Ghahramani, 2005; Ding, Xiang, Molloy, Li et al., 2010), and three proposed models (GP-dIBP-NMF, BB-dIBP-NMF, and C-dIBP-NMF). When applied to document-word co-clustering, the output A and X from each above model could be considered as the new representations of documents and words on latent factors. Based on these new representations, a clustering algorithm (K-means can be used and, in this section, K is set as the number of benchmark clusters) to do documents and words clustering. Since a common algorithm (i.e., K-means) is adopted for all models, the performances of models will be only determined by the learned new data representation A and X. Before the analysis on the clustering results, it needs to compare the convergence rates of these models. GP-dIBP-NMF takes about 1100 iterations to get stable after the burn-in stage (i.e., first 1000 iterations) on cora dataset, while C-dIBP-NMF and BB-dIBP-NMF take 200 and 500 iterations respectively.

For citeseer dataset, GP-dIBP-NMF takes about 1600 iterations to get stable after the burn-in stage (i.e., first 1000 iterations), while C-dIBP-NMF and BB-dIBP-NMF take 400 and 700 iterations respectively. The efficiency is due to the relatively simple model structures of the proposed model comparing the GP-based model. The clustering results on *Citeseer* are shown in Fig. 3.7 and Fig. 3.8. Since SNMF needs the factor number as the input, this parameter is adjusted for this model (the x-axis of each subfigure). Except for SNMF, other models are all nonparametric models which do not need a factor number as an input, so the results of these models are also plotted as horizontal lines in the figure. The learned factor numbers are 9 (from sIBP-NMF), 10 (from BB-dIBP-NMF), 12 (from C-dIBP-NMF) and 12 (from GP-dIBP-NMF). The results on *Cora* are shown in Fig. 3.9 and Fig. 3.10. The learned factor numbers are 10 (from sIBP-NMF), 17 (from BB-dIBP-NMF), 15 (from C-dIBP-NMF) and 22 (from GP-dIBP-NMF). From these results, it can been seen that: 1) there are fluctuations on the performance of SNMF on documents and words clustering due to its factor number input. 2) although sIBP-NMF has relatively good performance on the documents clustering, the performance on the words clustering is really bad, which may be due to its single side sparse and nonparametric control. 3) Comparing sIBP-NMF, the proposed three models all have better performances on both documents and words clustering. To summarize, without the prior knowledge of the number of factors, the proposed algorithms achieve relatively good performance on the document clustering task. Of the three algorithms, BB-dIBP-NMF achieves the best performance in all.

3.6 Summary

Nonnegative matrix factorization is advantageous for many machine learning tasks (e.g., co-clustering), but the assumption that the dimension of the factors is known in advance makes NMF impractical for many applications. To resolve this issue, this chapter has proposed a nonparametric sparse NMF framework based on dIBP to remove the assumption. First, a model is built by implementing this framework using GP-based dIBP, which successfully removes the assumption but suffers from larger model complexity and less flexibility. Then, two new dIBPs has been proposed through a bivariate Beta distribution and a copula. Advantages of the models based on new dIBPs is that 1) they have simpler model structures than models with GP-based dIBP; 2) the correlation between data could be directly learned out which could be seen as a measurement of the focus degree of hidden factors/topics. Lastly, three inference algorithms have been designed for the proposed three models, respectively. The experiments on the synthetic and real-world datasets demonstrates the capability of the proposed models to perform NMF without predefining the dimension number, and more correlation flexibility comparing GP-based dIBP.

Chapter 4

Bayesian Nonparametric Deep Topic Model for the Topic Hierarchy Learning

4.1 Introduction

Any document collection is with a number of hidden topics as learned in Chapter 3. If the documents are scientific publications, then a topic may be a research field: *Computer Vision*, *Text Mining* or *Image Segmentation*. These topics can be seen as a summarisation or abstraction of the whole document collection. Apparently, the topics are with different granularity which is not considered in Chapter 3. For example, *Image Segmentation* is a sub-area of research field *Computer Vision*, so *Computer Vision* is more general than *Image Segmentation*. Obtaining different levels of summarisation or abstraction of a document collection (named topic hierarchy) is of practice for many real-world applications including but not limited to document retrieval, summarisation, browsing and visualization. This study only sticks to the document modeling task throughout the rest of this chapter, although the usage of the topic hierarchy can be easily extend to other domains (i.e., image processing and genetic analysis)(Blei, Griffiths and Jordan, 2010).

Considering the significance of topic hierarchy learning, it has attracted much attention of the researchers and then many state-of-the-art methods or techniques have been proposed to resolve this issue: hierarchical clustering algorithm-based (Chuang and Chien, 2004), linear discriminant projection-based (Li et al., 2003), fuzzy relations-based (joon Kim and goo Lee, 2003), bag-of-related-words-based (Rossi and Rezende, 2011), word relations-based (Zhu et al., 2013), and so on. These methods construct topic hierarchy through document or words clustering. This clustering strategy will make a hard assignment of documents to topics, and especially words-based clustering will make the different topics exclusive with each other. Except for these techniques, topic models (Blei et al., 2003; Blei, 2012) have been proofed a very successful tool for the document analysis in practice. But the traditional topic models only output a topic set without hierarchy structure. Some revisions have been made to learn the topic hierarchy based on the topic models, such as Tree Labeled LDA (Slutsky et al., 2013), Hierarchically Labeled LDA (Perotte et al., 2011), Pachinko Allocation (Li and McCallum, 2006; Mimno et al., 2007) and fixed structure LDA (Reisinger and Paşca, 2009). However, these extensions suffer from one drawback: the number of topics need to be prefixed.

This chapter proposes a nonparametric deep topic model to learn the topic hierarchy from a document collection based on idea of hierarchically dependent random probability measures. The similar idea has already been used in Hierarchial Dirichlet Processes (HDP) (Teh, 2006), but the topics at upper layer will be 'copied' to the lower layer in HDP. This is not a desired phenomenon, and it hopes the topics at different layers will be different topics with different granularity. Therefore, a new hierarchical dependence strategy between two random probability measures has been designed through constructing a Dirichlet Mixture Probability Measure based on the random measure at the upper layer. Since this new designed measure is a continuous measure, the probability of 'copy' behavior is equal to zero. At the same time, this new designed measure takes the upper topics as the modes, so the topics at lower layer will be around the these modes with big probability. Another advantage of this model is to allow the number of topics in each layer could be automatically dynamic adjusted according to the input documents because of the nonparametric model nature. However, most of the current nonparametric models (i.e., nested Chinese Restaurant Process (nCRP) (Blei, Griffiths and Jordan, 2010) and nested Hierarchial Dirichlet Processes (nHDP)) use truncation-based method as an approximation. In order to keep this flexibility, a slice sampling inference algorithm which can be seen as an adaptive truncation method is designed to learn the topics and their hierarchical relations with the given documents. Experiments on the real-world document collections show the efficiency of the proposed model on the topic hierarchy learning.

The rest of this chapter is organized as follows: Section 4.2 gives the basic definitions of some concepts and states the problem formally; Section 4.3 introduces the proposed model in detail; Section 4.4 presents the inference algorithm and parameter estimation; Section 4.5 evaluates the proposed model through experiments; and Section 4.6 summarizes this chapter.

4.2 Problem statement

Given a document collection $\mathcal{D}$, it would be helpful to find the word patterns in $\mathcal{D}$ to assist further document analysis, such as document clustering, classification, visualization. These word patterns are commonly named ‘Topics’.

Definition 3 *A* Topic *of a corpus $\mathcal{D}$ is a normalized vector with length of the vocabulary size.*

Definition 4 *A* Topic Hierarchy *of a corpus $\mathcal{D}$ is a hierarchical structure consisted by different layers with a number of topics as nodes in each layer, and there are also links between nodes/topics in neighboring layers. It has the following properties:*

- *the topics at each layer are a summarization of the corpus;*
- *the topics at upper layers are more general than the ones at lower layers;*
- *the number of topics at upper layers are less than the ones at lower layers;*
- *each topic has one and only one parent topic at the neighboring upper layer.*

Without prefixing the number of layers, the limit of the top layer will only contain one topic considering the properties of the Topic Hierarchy. An example of Topic Hierarchy is shown in Fig. 4.1. The final aim of this chapter is to automatically learn this Topic Hierarchy from document collection.

4.3 Bayesian nonparametric deep topic model

This section proposes a new Bayesian nonparametric model to learn a topic hierarchy from a corpus. The basic ingredient of the proposed model is Dirichlet Process,

Definition 5 *A* Dirichlet Process (DP) *(Teh, 2010), which is specified by a base measure H on a measurable space Θ and a concentration parameter α, is a set of countably infinite random variables that can be seen as the measures on partitions*

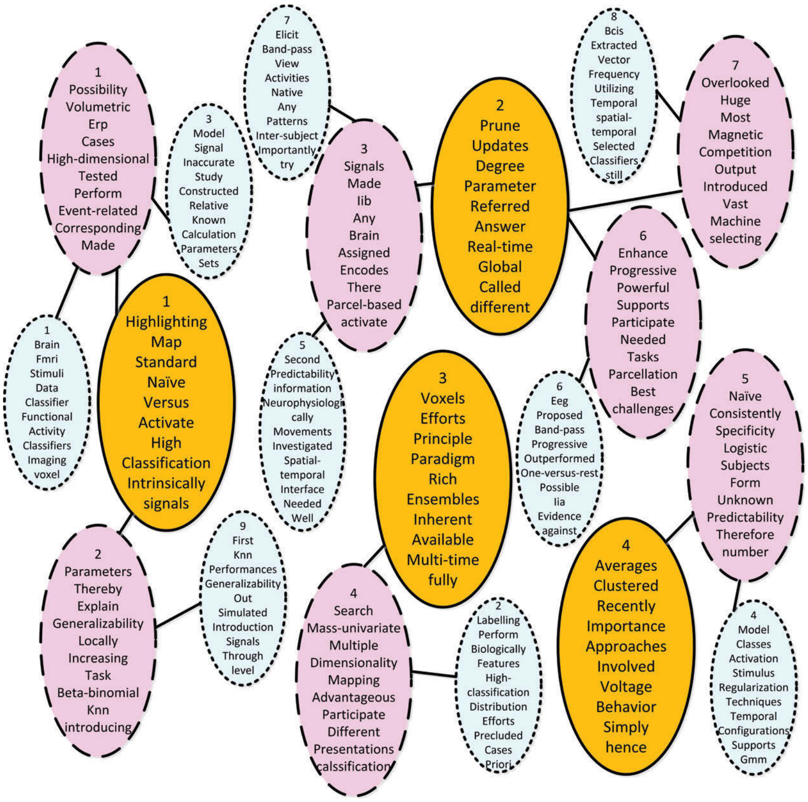


Figure 4.1: An example of Topic Hierarchy. The orange eclipses denote 4 topics at the first layer; the pink eclipses denote 7 topics at the second layer; the cyan eclipses denote 9 topics at the third layer. Data: abstracts of 11 papers from the special issue on Brain Encoding in *Pattern Recognition Journal*. After removing some stopwords, there are total 635 different words and each abstract contains about 100 words. α function: $y = 5x$ and λ function is $y = 5x$.

from a random infinite partition $\{\Omega\}_{k=1}^{\infty}$ of Θ. For a finite partition $\{\Omega\}_{k=1}^{K}$, the variables (measures on these partitions) from DP satisfy a Dirichlet distribution parameterized by the measures from based measure H on relative partitions

$$G(\Omega_1), \ldots, G(\Omega_K) \sim Dir(\alpha H(\Omega_1), \ldots, \alpha H(\Omega_K)) \tag{4.1}$$

where G is a realization of $DP(\alpha, H)$ and Dir denotes the Dirichlet distribution.

Furthermore, for each realization G of a DP, it can be equally expressed as

$$G \sim DP(\alpha, H) \Longrightarrow G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}, \quad \theta_k \sim H \tag{4.2}$$

where π_k ($\sum_k \pi_k = 1$) is the stick weights on the atom θ_k, $\delta_{\theta_k}()$ is an unit mass function, i.e., $\delta_{\theta_k}(\hat{\theta}) = 1$ if $\hat{\theta} = \theta_k$; $\delta_{\theta_k}(\hat{\theta}) = 0$, otherwise, and $\{\pi_k\}$ can be explicitly constructed through a stick-breaking process(Sethuraman, 1994). From Eq. 4.2, it can be found that G can also be seen as a purely discrete measure on the measurable space Θ concentrating on countable infinite points.

For the documents modeling task, the measurable space Θ is the N-simplex where N is the total number of words in the documents. Θ can also be simply seen as a set of all the topics. $\{\theta\}_k^{\infty}$, which can be seen as a countable infinite points on the N-simplex, denote a countable infinite number of topics. Simultaneously, $\{\pi\}_k^{\infty}$ denote the weights on the corresponding topics. Therefore, when assigned to a document, a $G_d = \sum_{k=1}^{\infty} \pi_{d,k} \delta_{\theta_k}$ denotes all the discussed topics and their corresponding weights in this document.

The discrete nature of the realization G from DP has been successfully utilized by Hierarchial Dirichlet Process (Teh, 2006) to share the same set of topics across different documents as

$$G_0 \sim DP(\alpha, H), \quad G_d \sim DP(\alpha, G_0) \tag{4.3}$$

where G_0 has defined a countable infinite set of topics and is considered as the based measure of the DP at the document level. Through this way, each G_d of a document can and only can inherit some or all of them. This topic sharing property is very useful in some tasks, but it is not desirable for the task: topic hierarchy learning. Considering the case that HDP is used to learn topic hierarchy, the topics in the top level will be inherited and emerge in the following

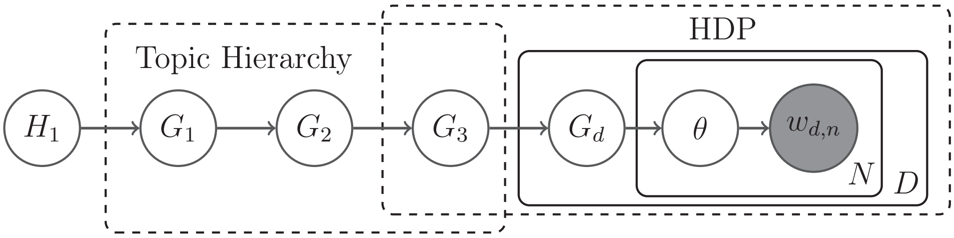


Figure 4.2: Graphical model for the generation of the proposed Deep Topic Model with three level Topic Hierarchy. The circles denotes random measures or random variables and the solid box denotes the repeat.

levels (of course, with different weights in each level). Apparently, it does not meet the objective that is to learn different granular topics in the topic hierarchy.

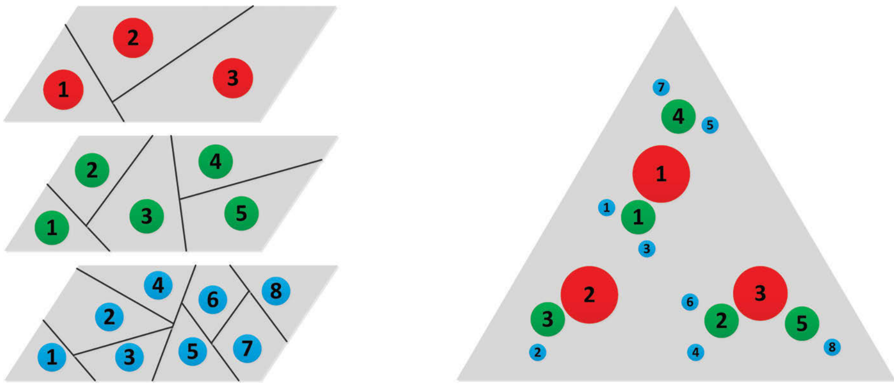


Figure 4.3: Illustration of deep random partition and the relations between topics on the simplex.

This chapter proposes a new nonparametric model to learn the topic hierarchy based on a new designed inherit strategy. In order to use the realization from DP as the base measure of the following layer, a new measure based on a realization from DP is designed,

Definition 6 *A* Dirichlet Mixture Probability Measure *(DMPM) is a continuous probability measure on a measurable space* Θ *which is constructed based on* *a realization from DP as*

$$H(\lambda, G) = \sum_{k=1}^{\infty} \pi_k Dir(\lambda\theta_k + 1), \qquad G = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k} \tag{4.4}$$

where G is a realization of $DP(\alpha, H)$ and Dir denotes the Dirichlet distribution.

Corollary 1 *As the increasing of the value of λ, the $H(\lambda, G)$ will be close to G.*

$$\lim_{\lambda \to +\infty} H(\lambda, G) = G \tag{4.5}$$

Proof 1 *Start with the limitations of the expectations and variances of the components of DMPM: Dirichlet distributions. The expectation of the Dirichlet distribution $Dir(\lambda\theta_k + 1)$ is $\frac{\lambda\theta_{k,v}+1}{\sum_{v=1}^{V}\lambda\theta_{k,v}+1}$, where V is the dimension of this Dirichlet distribution (i.e., the vocabulary size for the topic model). Since θ_k is a topic from another Dirichlet distribution, it is with the property: $\sum_{v=1}^{V}\theta_{k,v} = 1$ and $0 \leq \theta_{k,v} \leq 1$. Therefore, the expectation $\frac{\lambda\theta_{k,v}+1}{\sum_{v=1}^{V}\lambda\theta_{k,v}+1}$ is equal to $\frac{\lambda\theta_{k,v}+1}{\lambda+V}$. Furthermore, when λ goes to infinity, the expectation goes to*

$$\lim_{\lambda \to +\infty} \frac{\lambda\theta_{k,v} + 1}{\lambda + V} = \theta_k.$$

Similarly, the variance of the Dirichlet distribution $Dir(\lambda\theta_k + 1)$ is

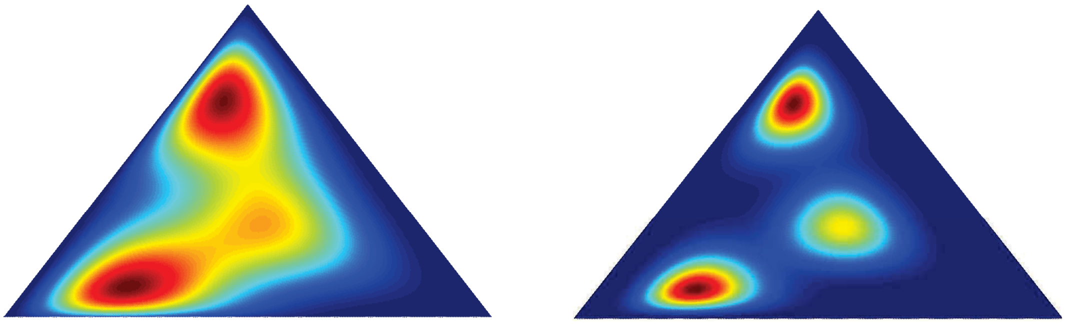
$$\frac{(\lambda\theta_{k,v} + 1)[\sum_{v=1}^{V}(\lambda\theta_{k,v} + 1) - (\lambda\theta_{k,v} + 1)]}{[\sum_{v=1}^{V}(\lambda\theta_{k,v} + 1)]^2([\sum_{v=1}^{V}(\lambda\theta_{k,v} + 1)] + 1)}$$

When λ goes to infinity, the variance goes to

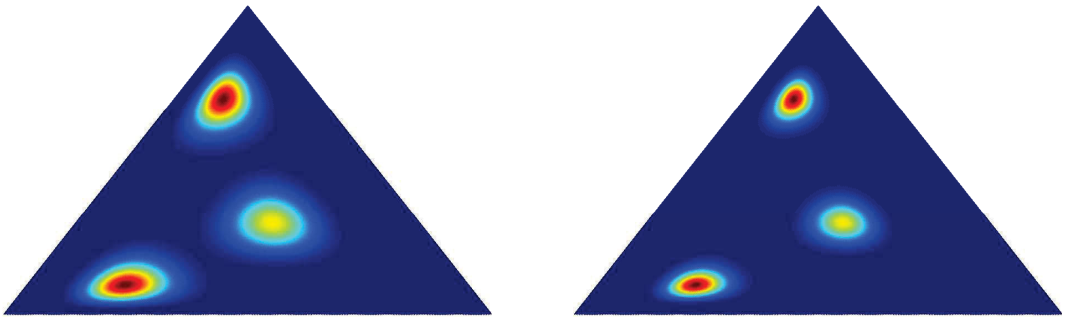
$$\lim_{\lambda \to +\infty} \frac{(\lambda\theta_{k,v} + 1)[\sum_{v=1}^{V}(\lambda\theta_{k,v} + 1) - (\lambda\theta_{k,v} + 1)]}{[\sum_{v=1}^{V}(\lambda\theta_{k,v} + 1)]^2([\sum_{v=1}^{V}(\lambda\theta_{k,v} + 1)] + 1)} = 0.$$

The zero variance means that the mass of the distribution totally concentrates to *the expectation θ_k. So, the distribution $Dir(\lambda\theta_k + 1)$ becomes δ_{θ_k}, and then the Eq. 4.5 is proofed.*

The DMPM has broken the discrete nature of the G. On the measurable space Θ, G has only countable infinite points with mass, but these points are only modes for the $H(\lambda, G)$. The λ is a mass concentration parameter. In other words, λ controls mass diffusion from the modes. As Fig. 4.4 shown, the larger λ leads to small mass concentration around modes.



(a) $\lambda = 10$ (b) $\lambda = 30$



(c) $\lambda = 60$ (d) $\lambda = 120$

Figure 4.4: The effect of the parameter λ for the DMPM. Here, G is a mixture of finite number (three) of Dirichlet distributions with equal weights, and each subfigure is an illustration of the $H(\lambda, G)$ under different value of λ.

Note that Dirichlet Mixture Model (Ma et al., 2014) (the finite version of the DMPM) has been investigated and applied to text modeling (Blei, 2004) and image processing (Bouguila et al., 2004). The difference between DMM and DMPM is that DMPM is composed by countable infinite number of Dirichlet distributions but DMM is composed by a finite number. This nature is important for the proposed nonparametric model, because it can model possible infinite number of topics. Given a document set, this model could get proper number of topics at each layer. The Infinite Mixture of Infinite Gaussian Mixtures (Yerebakan et al., 2014) is similar with us, as they both use HDP as a framework, and try to convert a discrete measure back to a continuous measure. However, the proposed approach uses the entire measure G, whereas their work uses just a single atom.

The complete Deep Topic Model (DTM) (here this aim is to learn a topic hierarchy with three layers and extending to topic hierarchies with more layers is easy) is defined as follows. The first layer is

$$\begin{aligned}
\beta &= [1, 1, \ldots, 1] \\
H_1 &= Dir(\beta) \\
G_1 &= DP(\alpha(1), H_1) \\
G_1 &= \sum_{k=1}^{\infty} \pi_k^{(1)} \cdot \delta_{\theta_k^{(1)}} \\
\pi_k^{(1)} &\sim GEM(\alpha(1)) \\
\theta_k^{(1)} &\sim H_1
\end{aligned}$$

where H_1 is a measure on N-simplex and the total mass is uniformly-distributed on this space under H_1 which is shown in Fig. 4.5(a). This base measure is used as a non-informative prior for the topics at the first layer when there is no data. G_1 is a realization from a DP with H_1 as base measure in which $\theta_k^{(1)}$ is a topic/node at the first level. The second layer is

$$
\begin{aligned}
H_2(\lambda(2), G_1) &= \sum_{k}^{\infty} \pi_k^{(1)} \cdot Dir(\lambda(2)\theta_k^{(1)} + 1) \\
G_2 &= DP(\alpha(2), H_2) \\
G_2 &= \sum_{i=1}^{\infty} \pi_i^{(2)} \cdot \delta_{\theta_i^{(2)}} \\
\pi_i^{(2)} &\sim GEM(\alpha(2)) \\
\theta_i^{(2)} &\sim H_2
\end{aligned}
$$

where H_2 is a Dirichlet mixture probability measure parameterized by $\lambda(2)$ and G_1.

Different from HDP, H_2 (One example is shown in Fig. 4.5(b)) is used as the base measure for the second layer DP rather than G_1. Considering the the continuous nature of the DMPM, the topics $\theta_i^{(2)}$ at this layer will be (with probability one) different from the ones θ_k in the first layer, and they are furthermore also different from each other. At the same time, these topics will surround the topics (the modes of the $H_2(\lambda(2), G_1)$) at the upper layer. Each new topic has its own variance (departure) from the modes. Therefore, these topics can be seen as the specified topics in a domain defined by the modes/topics. One example is that the modes/topics in the first layer are: *text mining* and *computer vision*. The topics in the second layers may be *document clustering*, *text summarization*, *image segmentation*, *multi-object tracking*. It can be seen that the topics at the lower layer will be more fine-grained comparing the ones in the upper layers, which just satisfies desired property of the Topic Hierarchy. Another interesting thing is the relations between the topics at the lower layer and the ones at the upper layer. Apparently, discovering their relations could help us find the inheriting relation between them. In the upper example, the topic *document clustering* and *text summarization* can be seen as the specification of the topic *text mining*; the topic *image segmentation* and *multi-object tracking* can be seen as the specification of the topic *computer vision*. This relation is left to the following Section.

The third layer is

$$
\begin{aligned}
H_3(\lambda(3), G_2) &= \sum_{i}^{\infty} \pi_i^{(2)} \cdot Dir(\lambda(3)\theta_i^{(2)} + 1) \\
G_3 &= DP(\alpha(3), H_3) \\
G_3 &= \sum_{j=1}^{\infty} \pi_j^{(3)} \cdot \delta_{\theta_j^{(3)}} \\
\pi_j^{(3)} &\sim GEM(\alpha(3)) \\
\theta_j^{(3)} &\sim H_3
\end{aligned}
$$

The idea is similar with the second layer, and one example is shown in Fig. 4.5(c). It needs to highlight again that the topics in the third layer will be more fine-grained than the ones in the second layer. Until now, we have obtained a prior for the three-layer topic hierarchy: $G_1 \to G_2 \to G_3$.

In order to learn this topic hierarchy from the document set, it needs to define the likelihood of the topic hierarchy prior. The requirements of the likelihood are: 1) different documents should share the same set of topics $\theta_j^{(3)}$ in the third layer; 2) the words in the same document should share the same set of topis; 3) the number of topics within one document should be not too much. Considering these requirements, we cannot directly use the G_3 as the prior for all the documents because it may make one document contain too many topics and different documents may not share any topics. Therefore, use the traditional HDP to link the G_3 to the documents. Each document is generated by

$$\begin{aligned}
G_d &= DP(\alpha_d, G_3) \\
G_d &= \sum_{t=1}^{\infty} \pi_t^d \cdot \delta_{\theta_t^d} \\
\pi_t^d &\sim GEM(\alpha_d) \\
x_{d,t} &\sim \pi^{(3)} \\
z_{d,n} &\sim \pi^d \\
w_{d,n} &\sim \theta_{x_{d,z_{d,n}}}
\end{aligned}$$

Each document only inherits a number of topis in the third layer $\theta_j^{(3)}$. The complete Deep Topic Model is graphically shown in Fig. 4.2.

In the DTM, $\lambda()$ and $\alpha()$ are two increasing functions with layer index as inputs. $\lambda()$ function is used to control the *Concentration* of the learned Topic Hierarchy. With the increase of the λ, the topics in the lower layer will tend to have smaller variance to the modes of the DMPM comparing the ones in the higher layer. The lower layer will tend to have more partitions comparing to higher layer. α function is used to control the *Width* of the learned Topic Hierarchy. With the increase of the α, the number of nodes/topics in the lower layers will be larger than the upper layers.

When considering each layer alone in Topic Hierarchy, we can seen it as a Chinese Restaurant Process (another representation of DP)[1]. Suppose the number of topics in the third layer is fixed as T_3. Considering the expectation of CRP, the number of topics in the second layer will be proportional to $T_2 \propto \log T_3$, and furthermore the number in the first layer will be proportional to $T_1 \propto \log \log T_3$. Apparently, the number of topics in the lower layer will tend to be larger than

[1] http://en.wikipedia.org/wiki/Chinese_restaurant_process

the number in the upper layer, which just meets the desired property of Topic Hierarchy. In the extreme case, there is infinite number of layers in the topic hierarchy, and then the number of topics at the first layer will be equal to one.

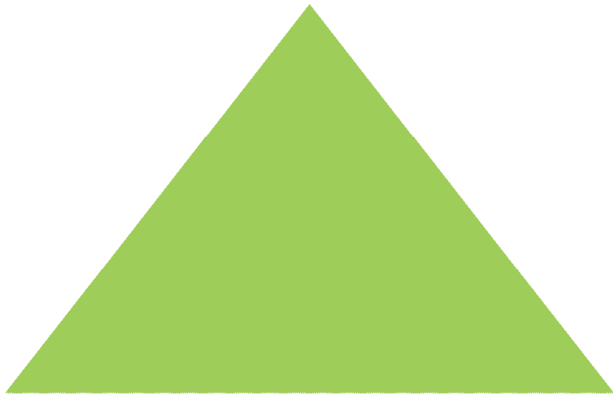
4.4 Model inference

With the designed Deep Topic Model and a document set, the goal of model inference is to estimate the latent variables in the model assuming the documents are generated according to generative procedure of the model. The definition of the model has actually defined a joint distribution of hidden variables and observations. Given the observations (i.e., documents for DTM), we are interested in the conditional distribution of the latent variables conditioned on the observation. For the proposed model, the distribution is

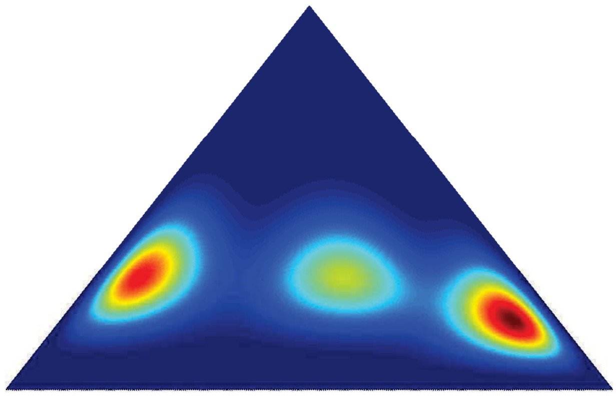
$$p(\pi^{(1)}, \pi^{(2)}, \pi^{(3)}, \pi_d, \theta^{(1)}, \theta^{(2)}, \theta^{(3)}, x_i, x_j, x_{d,t}, x_{d,n}|\mathcal{D}) \tag{4.6}$$

Normally, this conditional distribution is complicated, high-dimensional, and intractable, so some approximation methods should be adopted to obtain its quantiles. Here, a Markov Chain Monte Carlo (MCMC) algorithm is developed to approximate the posterior distribution of latent variables, which has been widely used in the probability graphical model area (Blei, 2004; Blei, Griffiths and Jordan, 2010; Teh, 2006). Through the properly designed MCMC algorithm, a bunch of samples of the target posterior distribution of latent variables can be obtained after a burn-in stage (a number of iterations). With the obtained samples, the quantiles of the target distribution can be estimated.

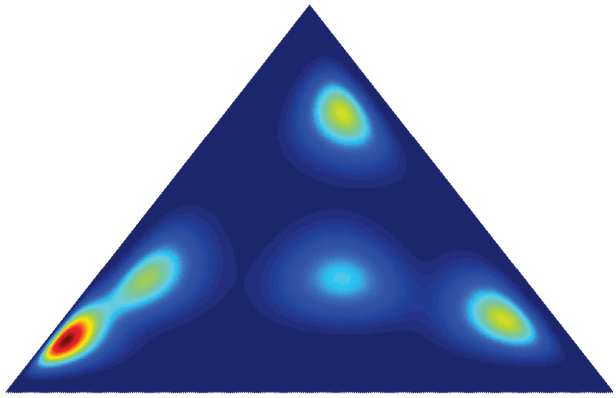
However, the infinite number of topics and their weights make the posterior inference of the latent variables even harder. One common work-around solution in nonparametric Bayesian learning is to use a truncation method. Truncation



(a) Base measure H_1 of the first layer



(b) Base measure H_2 of the second layer



(c) Base measure H_3 of the third layer

Figure 4.5: Illustration of the base measures in Deep Topic Model

method (Fox, 2009; Willsky et al., 2009) is widely accepted, which uses a relatively big K as the (potential) maximum number of topics. But there will be an approximation error brought from the truncation. Another successful technique to resolve this problem is developed: *Data/Variable Augmentation* (Van Dyk and Meng, 2001; Tanner and Wong, 2010), which is also known as *Slice Sampling* (Neal, 2003*b*; Kalli et al., 2011). The idea of this technique is to introduce additional variables to make the sampling of the posterior distribution more easy. For the Bayesian nonparametric model, this technique is even more important, which makes the posterior distribution with infinite number latent variables inferable. The difficult part is how to design the additional variables which is really model-dependent.

Next, a slice sampler for DTM is designed based on the technique developed for DP mixture model (Neal, 2003*b*; Kalli et al., 2011). The conditional distributions of the latent variables are given as follows,

Sampling $\mathbf{u^{(4)}_{d,n}}$. Since G_d contains countable infinite number of candidate topics to be assigned to the words in a document. $u^{(4)}_{d,n}$ is an introduced additional variable which can make the possible assigned topics to word n in document d within a finite scope, which is also known as *adaptive truncation*.

$$u^{(4)}_{d,n} \sim U(0, \pi^d_{t=x_{d,n}}) \tag{4.7}$$

where $t = x_{d,n}$ is the former assignment and $U(0, \pi^d_{t=x_{d,n}})$ denotes a uniform distribution with support on $[0, \pi^d_{t=x_{d,n}}]$.

Sampling $\mathbf{x_{d,n}}$. It is an indicator variable to express the assigned topic to word n in document d. The posterior conditional distribution is

$$p(x_{d,n} = t|\cdots) \propto \mathbb{1}(u^{(4)}_{d,n} < \pi^d_t) \cdot \theta^{(3)}_{x_{d,x_{d,n}},n} \tag{4.8}$$

where $\mathbb{1}()$ is an indicator function and equal to 1 if the condition is satisfied; otherwise, 0. The *adaptive truncation* is reflected by the sampling of the $x_{d,n}$. Only the topics with larger weights than $u^{(4)}_{d,n}$ are considered as the assignment candidates. Since the value of $u^{(4)}_{d,n}$ is larger than zero, this number must be finite.

Sampling $\pi^{\mathbf{d}}_{\mathbf{t}}$. The update of the stick weights of topics in a document should consider the upper adaptive truncation (i.e., the inequalities between slice variables and the stick weights). In (Neal, 2003*b*; Kalli et al., 2011), it has been proofed that the stick weights satisfy truncated Beta distributions. For $t \leq t^*$,

$$\nu^d_t \sim beta(\nu^d_t; 1, \alpha_d) \cdot \mathbb{1}(L_t < \nu^d_t < U_t) \tag{4.9}$$

where

$$\begin{aligned} t^* &= \max_n \{x_{d,n}\} \\ L_t &= \max_{x_{d,n}=t} \left\{ \frac{u^{(4)}_{d,n}}{\prod_{l<t}(1-\nu^d_l)} \right\} \\ U_t &= 1 - \max_{x_{d,n}>t} \left\{ \frac{u^{(4)}_{d,n}}{\nu^d_{x_{d,n}} \prod_{l<x_{d,n}, l\neq t}(1-\nu^d_l)} \right\} \end{aligned} \tag{4.10}$$

where t^* is the maximum number of used weights by the words in a document. Note that we could only update these weights (finite number) leaving the others (infinite number) un-updated, because the posterior of other weights are same with their priors.

Sampling $\mathbf{u^{(3)}_{d,t}}$. It is the slice variable for the stick weights in the third layer.

$$u^{(3)}_{d,t} \sim U(0, \pi^{(3)}_{j=x_{d,t}}) \tag{4.11}$$

Sampling $\mathbf{x_{d,t}}$. It is an indicator variable to express the assigned topic in the third layer to the t-th topic in document d. The posterior conditional distribution is

$$p(x_{d,t} = j | \cdots) \propto \mathbb{1}(u_{d,t} < \pi_j^{(3)}) \cdot \prod_{n \in d,t} \theta_{x_{d,t},n}^{(3)} \tag{4.12}$$

Note that the likelihood for $x_{d,t}$ is associated with all the words on the t-the topic.

Sampling $\pi_{\mathbf{j}}^{(\mathbf{3})}$. The update of the stick weights of topics in the third layer is similar with the ones in a document. One thing needs to be clarified is that the maximum number of updated weights should be determined by all the documents (i.e., all the inequalities for the words in all the documents should be considered). For $j \leq j^*$,

$$\nu_j^{(3)} \sim beta(\nu_j^{(3)}; 1, \alpha(3)) \cdot \mathbb{1}(L_j < \nu_j^{(3)} < U_j) \tag{4.13}$$

where

$$\begin{aligned} j^* &= \max_{d,t}\{x_{d,t}\} \\ L_j &= \max_{x_{d,t}=j} \left\{ \frac{u_{d,t}}{\prod_{l<x_{d,t}}(1-\nu_l^{(3)})} \right\} \\ U_j &= 1 - \max_{x_{d,t}>j} \left\{ \frac{u_{d,t}}{\nu_{x_{d,t}}^{(3)} \prod_{l<x_{d,t}, l \neq j}(1-\nu_l^{(3)})} \right\} \end{aligned} \tag{4.14}$$

Sampling $\theta_{\mathbf{j}}^{(\mathbf{3})}$. $\theta_j^{(3)}$ denotes a topic in the third layer, and its prior in the model is

$$p(\theta_j^{(3)}) = \sum_i^{\infty} \pi_i^{(2)} \cdot Dir(\lambda(3)\theta_i^{(2)}) \tag{4.15}$$

With the help of the indicator variable x_j, the posterior is

$$p(\theta_j^{(3)} | x_j = i, \cdots) \propto Dir(\theta_j^{(3)}; \lambda(3)\theta_i^{(2)}) \tag{4.16}$$

It can be seen that the documents are also associated in the posterior of the topics in the second layers. The reason is that it hopes that the topics in the

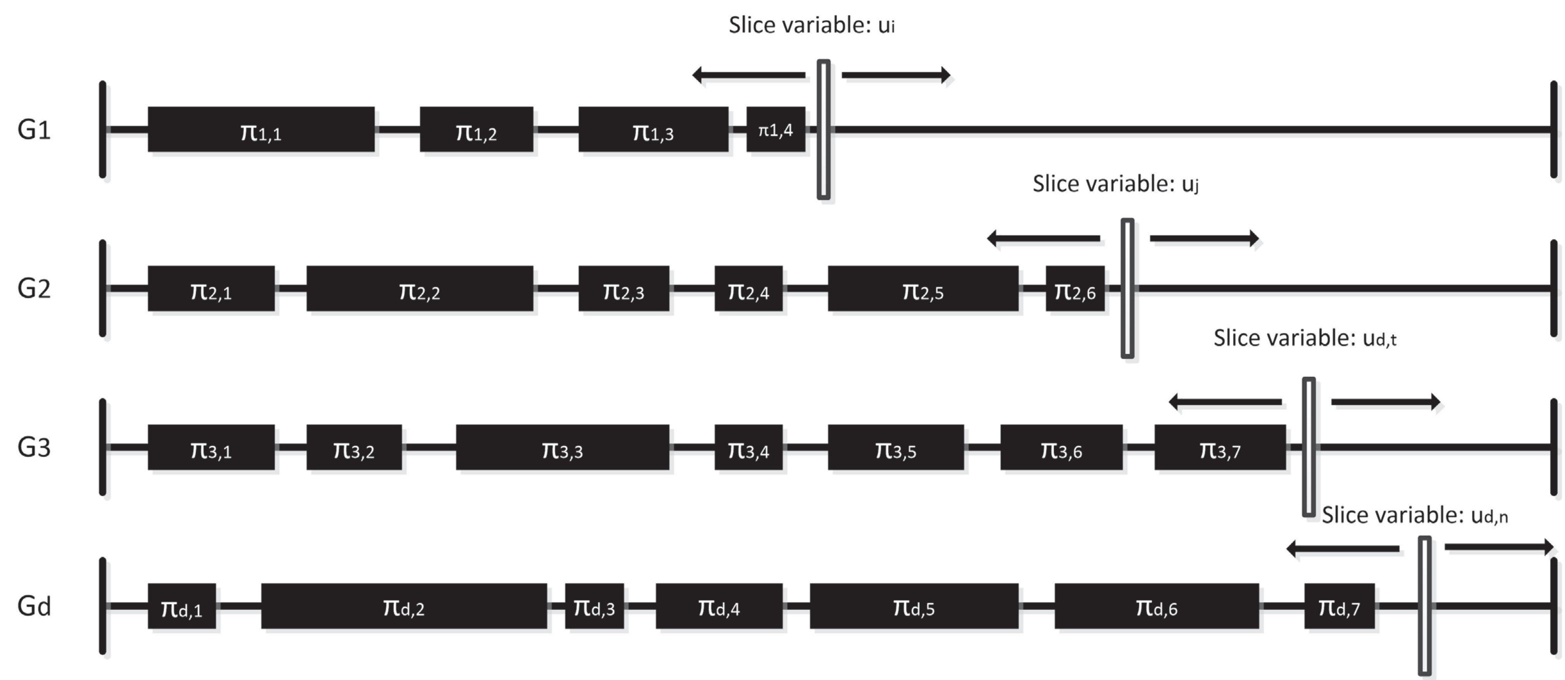


Figure 4.6: Illustration of the adaptive truncation by Slice variables. The whole length of a black line is one. Each green box denotes a stick weight, and the summation of the sticks on the same black line is one. The red bar represents a slice variable which functions as an adaptive truncation. At each iteration, the slices could move to both directions. With the slice variables, there are only finite number of stick weights and topis need to be updated.

upper layer will not only be the prior for the topics in the lower layer but also an summarization of the documents. For a topic in the second layer, the likelihood is not only the linked topics in the third layer but also the data associated with these topics.

Sampling $\mathbf{u_j^{(2)}}$. It is the slice variable for the stick weights in the second layer.

$$u_j^{(2)} \sim U(0, \pi_{i=x_j}) \tag{4.17}$$

Sampling $\mathbf{x_j}$. It is an indicator variable to express the assigned topic in the second layer to j-th topic the third layer. The posterior conditional distribution is

$$p(x_j = i|\cdots) \propto \mathbb{1}(u_j < \pi_i^{(2)}) \cdot Dir(\theta_j^{(3)}; \lambda(3)\theta_i^{(2)}) \tag{4.18}$$

It is worth to mention that the indicator x_j is not only for the ease of sampling but also gives a way to learn the links between the topics in the neighboring layers in the Topic Hierarchy. As stated in Definition 4, each topic/node in the topic hierarchy will have one and only one parent topic/node in the upper and neighboring layer. How to determine which topic is the parent topic for the nodes in a lower layer can be resolved through the designed indicators here. The links between the topics within different layers are also part of output of the model, which can facilitate the exploration of a documents using different granular levels, such as information retrieval task (Fersini et al., 2008).

Sampling $\pi_\mathbf{i}^{(\mathbf{2})}$. The update of the stick weights of topics in the second layer is similar with the ones in a document. For $i \leq i^*$,

$$\nu_i^{(2)} \sim beta(\nu_i; 1, \alpha(2)) \cdot \mathbb{1}(L_i < \nu_i^{(2)} < U_i) \tag{4.19}$$

where

$$i^* = \max_j \{x_j\}$$
$$L_i = \max_{x_j=i} \left\{ \frac{u_j^{(2)}}{\prod_{l<i}(1-\nu_l^{(2)})} \right\} \tag{4.20}$$
$$U_i = 1 - \max_{x_j>i} \left\{ \frac{u_j^{(2)}}{\nu_{x_j}^{(2)} \prod_{l<x_j, l\neq i}(1-\nu_l^{(2)})} \right\}$$

Sampling $\theta_{\mathbf{i}}^{(\mathbf{2})}$. $\theta_i^{(2)}$ denotes a topic in the third layer, and its prior in the model is

$$p(\theta_i^{(2)}) = \sum_k^{\infty} \pi_k^{(1)} \cdot Dir(\lambda(2)\theta_k^{(1)}) \tag{4.21}$$

With the help of the indicator variable x_i, the posterior is

$$p(\theta_i^{(2)}|x_i = k) \propto Dir(\theta_i^{(2)}; \lambda(2)\theta_k^{(1)}) \cdot \prod_{x_j=i} Dir(\theta_j^{(3)}|\lambda(3)\theta_i^{(2)}) \tag{4.22}$$

The likelihood of $\theta_i^{(2)}$ also contains the words assigned to all the topics at the third layer and linked to $\theta_i^{(2)}$.

Sampling $\mathbf{u}_{\mathbf{i}}^{(\mathbf{1})}$. It is the slice variable for the stick weights in the first layer.

$$u_i^{(1)} \sim U(0, \pi_{k=x_i}) \tag{4.23}$$

Sampling $\mathbf{x}_{\mathbf{i}}$. This indicator variable is to indicate the assigned topic in the first layer to i-th topic the second layer. The posterior conditional distribution is

$$p(x_i = k|\cdots) \propto \mathbb{1}(u_i < \pi_k^{(1)}) \cdot Dir(\theta_i^{(2)}; \lambda(2)\theta_k^{(1)}) \tag{4.24}$$

Sampling $\pi_{\mathbf{k}}^{(\mathbf{1})}$. For $k \leq k^*$,

$$\nu_k^{(1)} \sim beta(\nu_k; 1, \alpha(1)) \cdot \mathbb{1}(L_k < \nu_k^{(1)} < U_k) \tag{4.25}$$

where

$$
\begin{aligned}
k^* &= \max_i \{x_i\} \\
L_k &= \max_{x_i = k} \left\{ \frac{u_i^{(1)}}{\prod_{l<i}(1-\nu_l^{(2)})} \right\} \\
U_k &= 1 - \max_{x_i > k} \left\{ \frac{u_i^{(1)}}{\nu_{x_i}^{(1)} \prod_{l<x_i, l\neq k}(1-\nu_l^{(1)})} \right\}
\end{aligned}
\tag{4.26}
$$

Sampling $\theta_{\mathbf{k}}^{(\mathbf{1})}$. $\theta_k^{(1)}$ denotes a topic in the first layer, and its prior in the model is

$$p(\theta_k^{(1)}) = Dir(\beta) \tag{4.27}$$

The posterior is

$$p(\theta_k^{(1)} | \cdots) \propto Dir(\theta_k^{(1)}; \beta) \cdot \prod_{x_i = k} Dir(\theta_i^{(2)} | \lambda(2)\theta_k^{(1)}) \tag{4.28}$$

The likelihood for this topic needs to account for as a subtree of the topic hierarchy. Based on these conditional distributions, the full inference scheme is summarized in Algorithm 4.1. At the initialization stage, all the latent variables are randomly initialized. In each Gibbs sampling iteration, it iteratively samples each variable conditioned on the rest.

4.5 Experimental evaluation

This section presents the empirical evaluations of the proposed nonparametric deep topic model with synthetic and real-world data. Through synthetic data, the behavior of the model under different parameters is evaluated, including α function and λ function. Following that, the model has compared with other state-of-the-art topic models, i.e., LDA (Blei et al., 2003) and HDP (Teh, 2006), based on six real-world document datasets. Although it is trivial to extend to

Algorithm 4.1: Slice Sampler for DTM

```
Input: A corpus, 𝒟
Output: Topic Hierarchy
initialization;
while iter ≤ max_iter do
    for d = 1; d ≤ D do
        for n = 1; n ≤ N_d do
            Update u^(4)_{d,n} by Eq. 4.7;
        for n = 1; n ≤ N_d do
            Update x_{d,n} by Eq. 4.8;
        for t = 1; t ≤ t* do
            Update π^d_t by Eq. 4.9;
        for t = 1; t ≤ t* do
            Update u^(3)_{d,t} by Eq. 4.11;
    for d = 1; d ≤ D do
        for t = 1; t ≤ t* do
            Update x_{d,t} by Eq. 4.12;
    for j = 1; j ≤ j* do
        Update π^(3)_j by Eq. 4.13;
        Update u^(2)_j by Eq. 4.17;
    for j = 1; j ≤ j* do
        Update x_j by Eq. 4.18;
        Update θ^(3)_j by Eq. 4.16;
    for i = 1; i ≤ i* do
        Update u^(1)_i by Eq. 4.23;
        Update x_i by Eq. 4.24;
        Update θ^(2)_i by Eq. 4.22;
    for k = 1; k ≤ k* do
        Update π^(1)_k by Eq. 4.25;
        Update θ^(1)_k by Eq. 4.27;
    iter++;
return {{θ_k}, {θ_i}, {θ_j}, {π_k}, {π_i}, {π_j}, x_i, x_j};
```

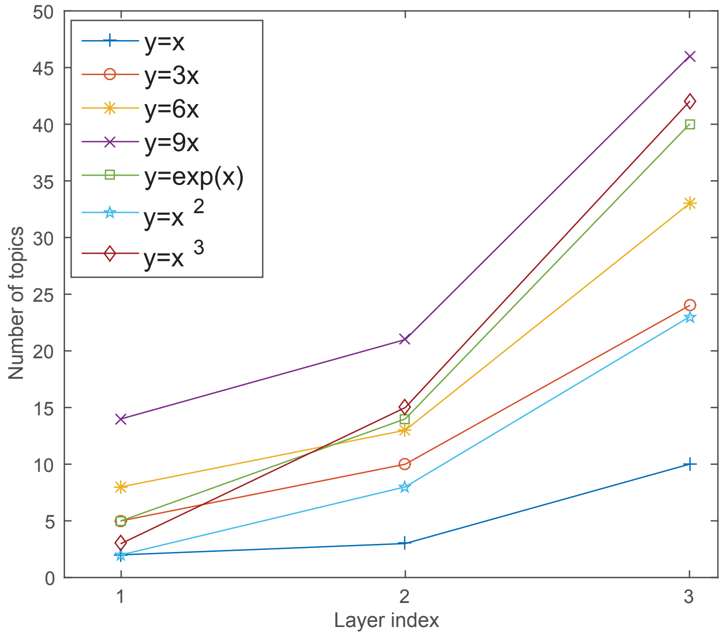


Figure 4.7: Illustration of the influence of the α function. The synthetic data is: 100 documents and 200 different words, and each document has 20 different words. The λ function is $y = x$.

more layers, there are only three layers learned by the model in this section. It needs to highlight that the number of topics in each layer is not fixed in advance and can be learnt from the data.

4.5.1 Synthetic data

As discussed in Section 4.3, the α function could control the number trending of topics in layers, and the λ function could control the concentration degree of topics in neighboring layers. In order to evaluate their influence, different function candidates for the α and λ function are compared. At first, randomly generate a document dataset with 100 documents and 20 words (vocabulary size). With λ function is fixed as $y = x$, then use the following function candidates for

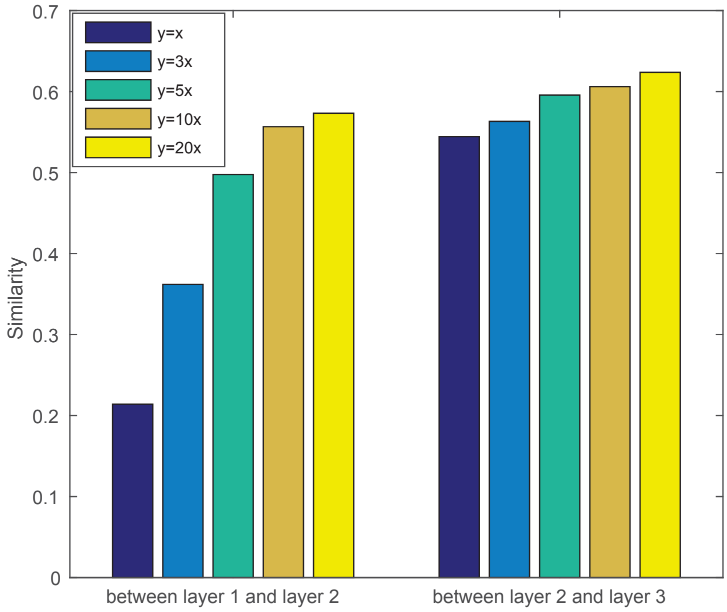


Figure 4.8: Illustration of the influence of the λ function. The synthetic data is: 100 documents and 200 different words, and each document has 20 different words. The α function is $y = x$.

α function: $y = x$, $y = 3x$, $y = 6x$, $y = 9x$, $y = \exp(x)$, $y = x^2$, and $y = x^3$. The results are shown in Figure 4.7. As seen from this figure, the number of topics are different with different α function, and the number of topics will increase with the value of α function. These results could give a hint to set the ratios between topic numbers on different layers. Similarly, different function candidates for the λ function are also compared, including $y = x$, $y = 3x$, $y = 5x$, $y = 10x$, and $y = 20x$, with α function fixed as $y = x$. In order to quantitatively evaluate the concentration degree, the following metric is proposed

$$S = \frac{1}{N_{h,l}} \sum_{h \in H, l \in L, h \to l} \theta_h \odot \theta_l \tag{4.29}$$

where θ_h is a topic at the higher layer H, θ_l is a topic at the lower layer L, h is the parent topic of l, $N_{h,l}$ is the number of pairs with two layers, and $\odot$ denotes inner product between two topics. This metric is to compute the similarity of two topics within two layers with a relation. The large average similarities shows the large concentration degree of topics. The final results are show in Fig. 4.7. There are two parts in the figure: one denotes the similarity between the topics at the first and second layers; the other denotes the similarity between the topics at the second and third layers. As the increasing of the coefficients of the linear functions, the λ at each layer will be also increase and then the topics at the lower layer will be more concentrated to the topics at the higher layer (the value of similarity is larger).

4.5.2 Real-world data

The efficiency of the proposed model (DTM) to explain the training documents and predict the testing documents is evaluated below. The selected comparative models are LDA[2] and HDP[3]. The real-world datasets used for evaluation are:

- **PR Dataset** The PR dataset consists of 11 abstracts from the special issue on Brain Encoding in *Pattern Recognition Journal*[4]. After removing some stopwords, there are total 635 different words and each abstract contains about 100 words.

- **ICDM Dataset** The ICDM dataset consists of 143 abstracts of all papers from *ICDM2014*[5] . After removing some stopwords, there are total 3664 different words and each abstract contains about 112 words.

[2]The implementation of LDA is from: http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm.

[3]The implementation of HDP is from the personal webpage of Prof. Yee Whye Teh: http://www.stats.ox.ac.uk/~teh/.

[4]http://www.sciencedirect.com/science/journal/00313203/45/6

[5]http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7022262

- **Cora Dataset**[6] The Cora dataset consists of 2708 scientific publications. Each publication in the dataset is described by a 0/1-valued word vector indicating the absence/presence of the corresponding word from the dictionary. The dictionary consists of 1433 unique words.

- **Citeseer Dataset**[6] The CiteSeer dataset consists of 3312 scientific publications. Each publication in the dataset is also described by a 0/1-valued word vector indicating the absence/presence of the corresponding word from the dictionary. The dictionary consists of 3703 unique words.

- **KOS Dataset**[7] The KOS dataset consists of 3430 documents with total 6906 words. The original source is *dailykos.com*.

- **NIPS Dataset**[7] The NIPS dataset consists of 1500 documents with total 12419 words. The original source is *books.nips.cc*.

For each dataset, the whole documents are randomly separated into two parts: training documents (around 80%) and test documents (around 20%). At first, run three models, i.e., DTM, LDA and HDP, on the training documents, and then predict the test documents using the trained models respectively. Note that there are two criteria: one is how well the model fits the training documents; the other is how well the model predicts the test documents. Considering that, the following commonly adopted metrics are used to evaluate them. The first one is

$$Lilkelihood = \sum_{d}^{D_{train}} \log p(w_d|model) \tag{4.30}$$

where D_{train} is the number of training documents and $p(w_d|model)$ is the probability of the document d given the model. If the Likelihood is larger, it denotes

[6] http://linqs.cs.umd.edu/projects/projects/lbc/
[7] https://archive.ics.uci.edu/ml/datasets/Bag+of+Words

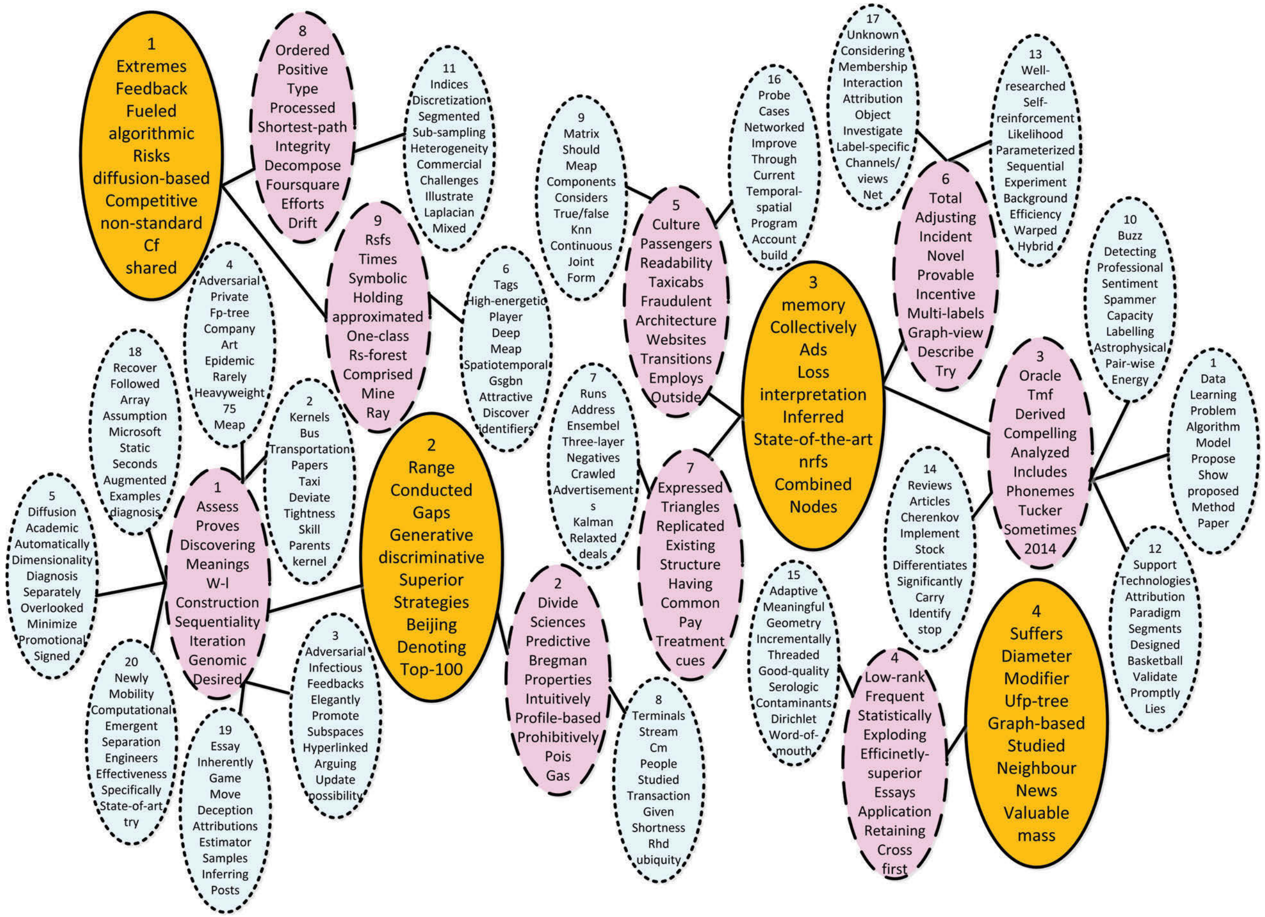


Figure 4.9: An example of topic hierarchy. The orange eclipses denote 4 topics at the first layer; the pink eclipses denote 9 topics at the second layer; the cyan eclipses denote 20 topics at the third layer. Dataset: *ICDM*. The α function is $y = 5x$ and the λ function is also $x = 5x$.

that the model fits the training documents better. For the test documents,

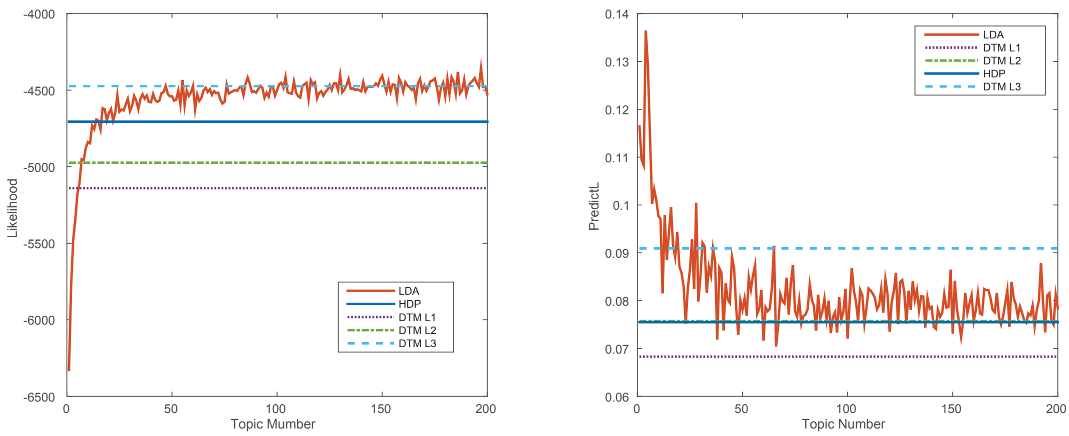
$$PredictL = \exp\left(-\frac{1}{D_{test}}\log p(w_d|model)\right) \tag{4.31}$$

where D_{test} is the number of test documents. This metric is also known as Perplexity used in other topic models. If the PredictL is smaller, it denotes that the model can better predicts the test documents. Since LDA needs the number of topics to be fixed in advance, topic number in LDA is adjusted within [1, 200]. In all experiments, the models run 1000 iterations with 200 burn-in stages.

Table 4.1: The learned topic numbers at three layers from all datasets

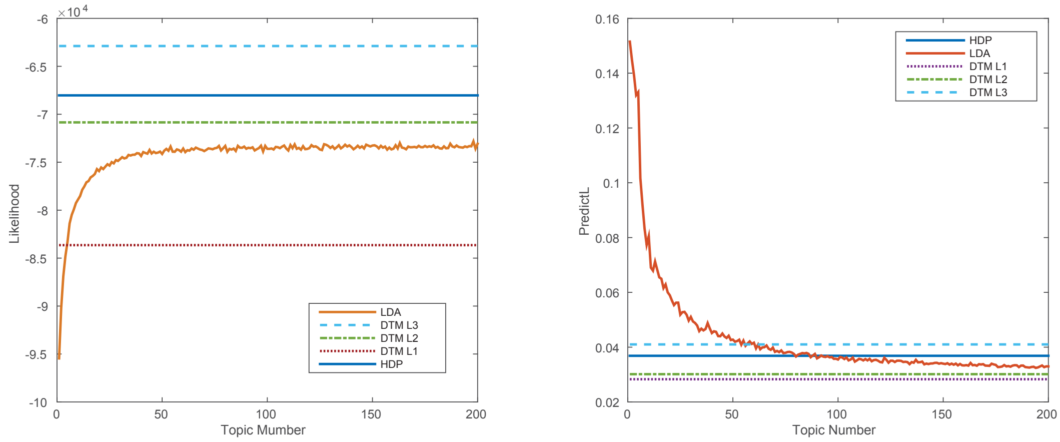
Dataset	HDP	DTM L1	DTM L2	DTM L3
PR	34	2	4	10
ICDM	73	6	9	16
CITESEER	22	5	11	30
CORA	12	3	7	27
KOS	18	5	21	41
NIPS	9	10	14	40

The results on all the datasets are repsectively shown in Fig. 4.10, Fig. 4.11, Fig. 4.12, Fig. 4.13, Fig. 4.14, and Fig. 4.15. Besides, another illustrative example for the learned Topic Hierarchy based on *ICDM* dataset is shown in Fig. 4.9. Each figure contains two subfiguers: one shows the training document Likelihood by Eq. 4.30; the other shows the test document PredictL by Eq. 4.31. Since the LDA needs to fix the topic number but HDP and DTM do not need, there is a curve for LDA where each point represents likelihood from LDA with the input topic number but only a straight line for HDP (HDP does not require the topic number as an input.) in the first subfiguer. For the proposed DTM, there are three layer topics and so there are three straight lines for DTM from three layer topics, because the topic number in each layer does also not need to be fixed for DTM. The learned topic number is listed in Table 4.1. From these results, it can be seen that the performance of HDP is as good as LDA on the training documents fitting but without prefixing the topic number. For the predicting test documents, HDP archives better results comparing LDA with an exception that they have comparative results on *CORA* and *CITESEER* datasets. Furthermore, it observes that the topics at lower layer have better performance than the topics at higher layer on the training documents fitting, but the topics at higher layer have better performance than the topics at lower layer on the test documents predicting. This is because that the topics in the higher layer are more general than the topics at the lower layer, so they have the ability to well generalize the training documents but the topics at lower layer have the ability to specific the training documents with more details. The better generalization ability help topics at the high layer achieves better performance on the predicting the unseen documents; the better specification ability help topics at the lower layer achieves better performance on the explaining the training documents.



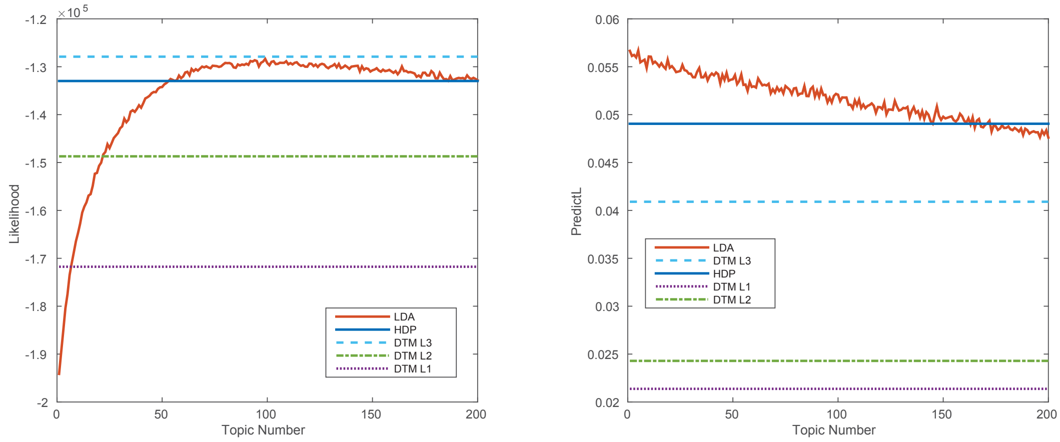
(a) *Likelihood* of the training documents (b) *PredictL* of the test documents

Figure 4.10: Results on Dataset *PR*



(a) *Likelihood* of the training documents (b) *PredictL* of the test documents

Figure 4.11: Results on Dataset *ICDM*

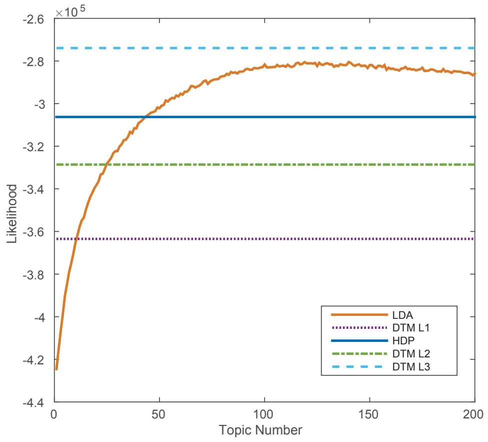


(a) *Likelihood* of the training documents (b) *PredictL* of the test documents

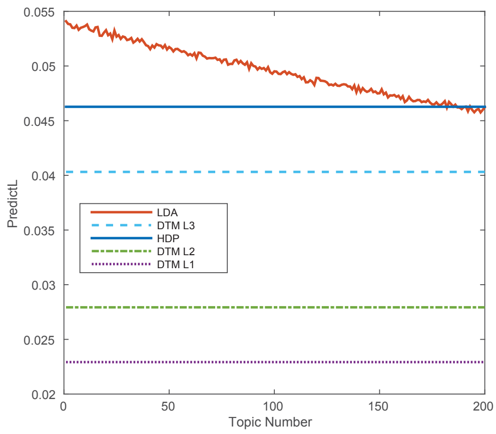
Figure 4.12: Results on Dataset *CORA*

4.6 Summary

Topic models are a kind of efficient tool for the document analysis through automatically learning out a set of topics. Comparing a topic set, a topic hierarchy

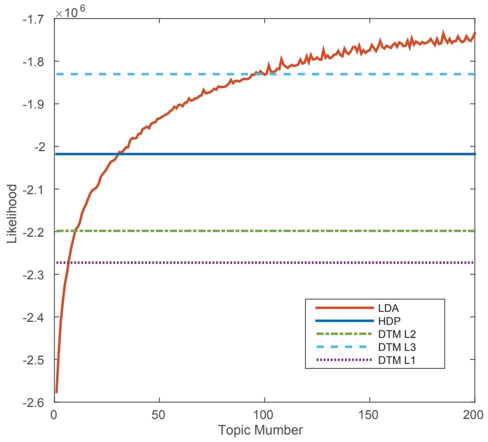


(a) *Likelihood* of the training documents

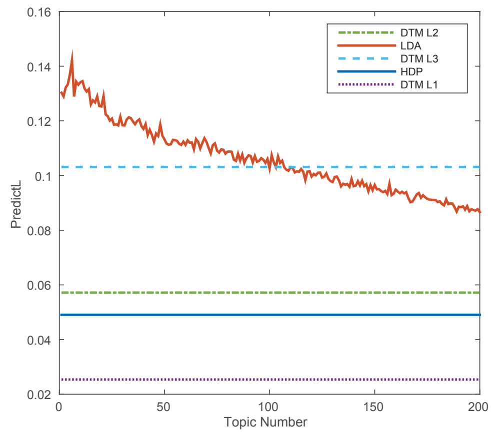


(b) *PredictL* of the test documents

Figure 4.13: Results on Dataset *Citeseer*



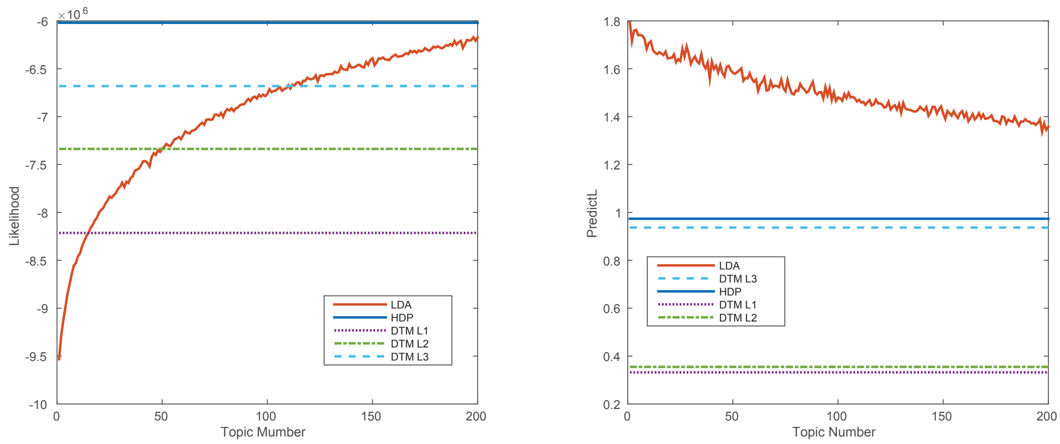
(a) *Likelihood* of the training documents



(b) *PredictL* of the test documents

Figure 4.14: Results on Dataset *KOS*

is more helpful for the corpus understanding and browsing. This chapter has proposed a nonparametric deep topic model to automatically learn out a topic hierarchy from any document collection without the requirement to prefix the number of topics at each layer. Inspired by the renowned Hierarchical Dirichlet



(a) *Likelihood* of the training documents (b) *PredictL* of the test documents

Figure 4.15: Results on Dataset *NIPS*

Processes, a new probability measure has been proposed, which has not only inherited the mechanisms of linking the topics at different layers but also broken the topic sharing strategy of the original HDP. By the help of this new probability measure, a topic hierarchy can be learned out. Besides, this chapter has also designed a slice sampling algorithm without the truncation approximation that is commonly used by many nonparametric models. The experimental results show that the topics at higher layer are more general than the topics at the lower layer, which have better ability to predict the new documents. On the contrary, the topics at lower layer are more specific than the topics at the higher layer, which have better ability to explain the training documents.

Chapter 5

Bayesian Nonparametric Relational Topic Model for Document Network Learning

5.1 Introduction

Understanding a corpus (without any additional information) in Chapter 3 and 4 is significant for businesses, organizations and individuals for instance the academic papers of IEEE, the emails in an organization and the previously browsed webpages of a person. One commonly accepted and successful way to understand a corpus is to discover the hidden topics in the corpus (Blei et al., 2003; Blei, 2012). The revealed hidden topics could improve the services of IEEE, such as the ability to search, browse or visualize academic papers ; help an organization understand and resolve the concerns of its employees; assist internet browsers to understand the interests of a person and then provide accurate personalized services. Furthermore, there are normally links (additional information) between the documents in a corpus. A paper citation network (Guo et al., 2014) is an example of a document network in which the academic papers are linked by their citation relations; an email network (Klimt and Yang, 2004) is a document network in which the emails are linked by their reply relations; a webpage network (Park, 2003) is a document network in which webpages are linked by their hyperlinks. Since these links also express the nature of the documents, it is apparent that hidden topic discovery should consider these links as well. Therefore, this Chapter will consider more complicated situation than Chapter 3 and 4.

Similar studies focusing on the hidden topics discovering from the document network using some Relational Topic Models (RTM) (Chang and Blei, 2009; Chang et al., 2010; Chen, Zhu, Xia and Zhang, 2015) have already been successfully developed. Unlike the traditional topic models (Blei et al., 2003; Blei, 2012) that focus on mining the hidden topics from a document corpus (without links between documents), the RTM can make discovered topics inherit the document network structure. The links between documents can be considered as constrains of the hidden topics.

One drawback of existing RTMs is that they are built with fixed-dimensional probability distributions, such as Dirichlet, Multinomial, Gamma and Possion distribution, which require their dimensions be fixed before use. Hence, the number of hidden topics must be specified in advance, and is normally chosen using domain knowledge. This is difficult and unrealistic in many real-world applications, so RTMs fail to find the number of topics in a document network.

In order to overcome this drawback, this chapter proposes a Nonparametric Relational Topic (NRT) model in this chapter, which removes the necessity of fixing the topic number. When aiming to build a NRT for a document network, there are three challenges:

1. how to express the document interest on infinite number of topics? Instead of probability distributions, stochastic processes are adopted by the pro-

posed model to express the interest of a document on the 'infinite' number of topics. Stochastic process can be simply considered as 'infinite' dimensional distributions[1].

2. How to make all the documents share the same set of topics? This is a common feature found in many real-world applications, and many literatures (Chang et al., 2010; Chen, Zhu, Xia and Zhang, 2015) have exploited this property in their work. In order to achieve the above requirement, this study has carefully designed the following mechanism: use a global Gamma process to represent a set of base components and each document has its own Gamma process thinned from them. The thinned Gamma processes help documents share the same set of topics. This is important because users are not interested in analyzing documents in a database without sharing any common topics (Teh, 2006).

3. How to make two linked documents have similar topics? This chapter handles this challenge by controlling the subsampling probabilities of all the documents on topics, and make the linked documents subsample the similar topics. A subsampling Markov Random Field is proposed as the model constraint.

Finally, two sampling algorithms are designed to learn the proposed model under different conditions. Experiments with document networks show some efficiency in learning what the hidden topics are and superior performance the modeląŕs ability to learn the number of hidden topics. It is worth noting that, although using document networks as examples throughout this chapter, this work can be applied to other networks with node features.

[1]We only consider the pure-jump processes in this study. Some continuous processes cannot be simply considered as the 'infinite' dimensional distributions.

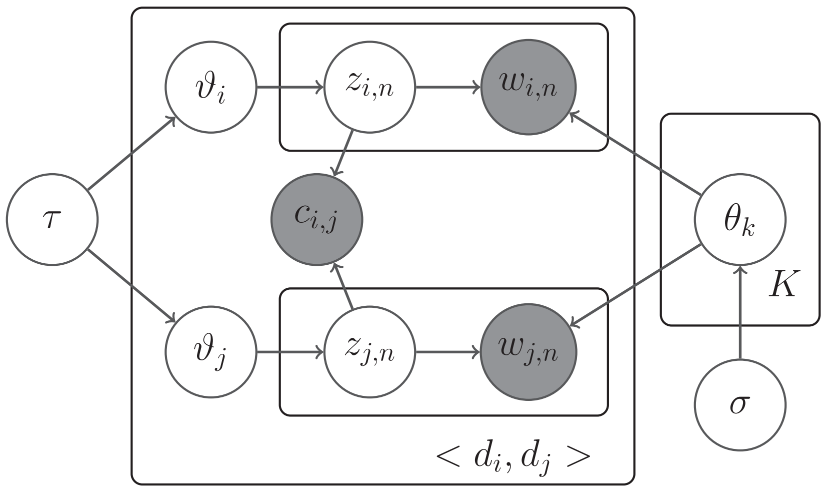


Figure 5.1: Graphical model of relational-topic-Model

The rest of this chapter is structured as follows. Section 5.2 introduces the preliminary knowledge. The proposed model is presented in Section 5.3 and the detailed derivations of its sampling inference have been illustrated in Section 5.4. Section 5.5 presents experimental results both on the synthetic and real-world data. Finally, Section 5.6 summarises this chapter.

5.2 Preliminary knowledge

This section briefly introduces the related models which will be used in the rest of sections.

5.2.1 Relational topic model

Since the finite relational topic models (Chang et al., 2010) are the comparative model, it is given more details below. The corresponding graphical representation is shown in Fig. 5.1, and the generative process is as follows

$$\begin{aligned} \vartheta_d &\overset{i.i.d}{\sim} Dirichlet(\tau) \\ \theta_k &\overset{i.i.d}{\sim} Dirichlet(\sigma) \\ z_{d,n} &\sim Discrete(\vartheta_d) \\ w_{d,n} &\sim Discrete(\theta_{z_{d,n}}) \\ c_{d_i,d_j} &\sim GLM(z_{d_i}, z_{d_j}) \end{aligned} \tag{5.1}$$

where ϑ_d is the topic distribution of a document d, θ_k is the word distribution of topic k, $z_{d,n}$ is the topic index of n-th word in document d, $w_{d,n}$ is n-th observed word in document d. All these variables are same with the original LDA (Blei et al., 2003). The different and significant part is the variable c, which denotes the observed document link. This model uses a Generalized Linear Model (GLM) (McCullagh, 1984) to model the generation of the document links

$$p(c_{d_i,d_j} = 1) \sim GLM(z_{d_i}, z_{d_j}) \tag{5.2}$$

One problem with this model is that the number of topics needs to be pre-defined and for some real-world applications, this is not trivial.

5.2.2 Markov random field

Markov Random Field (MRF) (Li, 1995) is an undirected graphical model that involves a set of random variables with links (probability dependencies). MRF represents the joint distribution of all the random variables having Markov property. Suppose there are N random variables $\{x\}_{i=1}^N$ and their joint distribution is

$$p(\{x\}_{i=1}^N) = \prod_{C \in clique(G)} \psi(C) \tag{5.3}$$

where G is the graph structure from the dependencies of random variables in MRF and C is a clique of G. $\psi(C)$ is normally called *energy function*. The factorization in the Eq. (5.3) is due to the Markov property. The ability of MRF is to make the marginal distribution of each variable conditioned on its neighbors.

5.3 Bayesian nonparametric relational topic model

In this section, the proposed Nonparametric Relational Topic (NRT) model for the document network is presented in detail. When building a NRT, there are three challenges: i) How to express the document interest on infinite number of topics? ii) How to make all the documents share the same set of topics? iii) How to make two linked documents share similar topics? In the following, strategies to handle the above three challenges are introduced one by one. Some frequently used notations are summarised in Table 5.1.

Challenge 1 *How to express the document interest on infinite number of topics?*

When the topic number is prefixed, it is simply to draw a random variable from a fixed-dimensional probability distribution (such as Dirichlet distribution and Logit-normal distribution) as the topic interest of a document in the traditional topic models. However, the probability distributions have to be abandoned for the model building when the number of topics cannot be reasonably prefixed with enough prior knowledge. It makes traditional topic models built by probability distributions not work.

Here, this chapter uses a draw from a Gamma process to express the interest of a document on infinite hidden topics. A Gamma process $GaP(\alpha, H)$ is a

Table 5.1: Important notations in this chapter

Symbol	meaning in this chapter
D	the number of documents
V	vocabulary size
K	the number of topics
d	document index
v	word index
k	topic index
N_d	number of words in document d
H	the base probability measure of a Gamma process
α	concentration parameter of a Gamma process
Γ	a random draw/realization of a Gamma process
π_k	weight of the topic k
$\pi_{d,k}$	weight of the topic k in document d
Θ	parameter (keyword distribution) space
θ_k	parameter (keyword distribution) of topic k
r_k	binary (indicator) variable of topic k
$r_{d,k}$	binary (indicator) variable of topic k in document d
$q_{d,k}$	the subsampling probability of document d keeping topic k
C	a clique of document network
$\psi(C)$	energy function on clique C
$n_{d,v}$	number of word v in document d
$n_{d,v,k}$	number of word v assigned to topic k in document d
$z_{d,v,m}$	topic index assigned to m-th word v in document d
$u_{d,v,m}$	auxiliary slice variable assigned to m-th word v in document d

stochastic process, where H is a base (shape) measure parameter on Θ and α is the concentration (scale) parameter. Let $\Gamma = \{(\pi_k, \theta_k)\}_{k=1}^{\infty}$ be a random draw of a Gamma process in the product space $\mathbb{R}^+ \times \Theta$ where $\pi_k \in \mathbb{R}^+$ and $\theta_k \in \Theta$, and it can be represented as

$$\begin{aligned}\Gamma &\sim GaP(\alpha, H)\\ &= \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}, \quad \theta_k \sim H\end{aligned} \tag{5.4}$$

where δ_{θ_k} is an unit mass function (i.e., $\delta_{\theta_k}(\theta^*) = 1$ if $\theta_k = \theta^*$ and $\delta_{\theta_k}(\theta^*) = 0$ if $\theta_k \neq \theta^*$); π_k satisfies an improper Gamma distribution $Gamma(0, \alpha)$ and that is why it is called *Gamma process*. Γ can also be seen as a complete random measure. More information about Gamma process can be found in (Rao and Teh, 2009; Roychowdhury and Kulis, 2015). When using Γ to express the document interest, the $\{\theta_k)\}_{k=1}^{\infty}$ in Eq. (5.4) denotes the infinite number of topics and $\{\pi_k)\}_{k=1}^{\infty}$ in Eq. (5.4) denotes the weights of infinite number of topics in a document. Note that π_k is within $(0, +\infty)$ not $[0, 1]$, but $\{\pi_k)\}_{k=1}^{\infty}$ can also be seen as the weights of topics in a document. As illustrated in Fig. 5.2, the idea is to assign each document a Gamma process. In this figure, each document is with a '*fence*' in which each bar has two properties: *position* that denotes the topic and *length* that denotes the weight of the corresponding topic in this document. Note that each document could set its fence positions at will. Due to the infinity of Γ, Challenge 1 can be handled for now.

Challenge 2 *How to make all the documents share the same set of topics?*

Considering the situation with infinite number of topics, it hopes that there are some topics that are shared by documents even with infinite number of candidate topics. Let us consider an extreme situation: each document in a document network is with and only with its own topics that are different from others. Apparently, this situation is not desirable because the motivation of the document modeling or topic models is to discover the shared knowledge (i.e. topics) of a document corpus.

Considering the continuity of the parameter space Θ (the *base line* of the *fence* in Fig. 5.2, equivalently), the probability that two documents are with same topics is 0. In order to handel Challenge 2, a global Gamma process is firstly generated as follow

$$\Gamma_0 \sim GaP(\alpha, H)$$

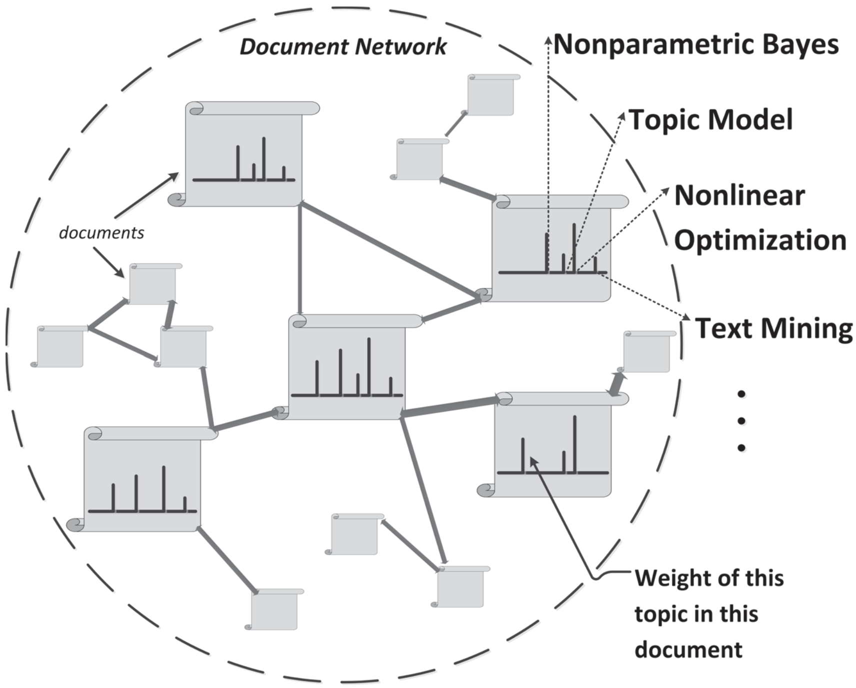


Figure 5.2: Illustration of Gamma process assignments for a document network. Each document is assigned a Gamma process which has infinite components (represented by the fences in the figure). Each fence denotes a hidden topic, and some examples are given in the figure. The length of the fences denote the weights of different topics in a document.

which is equal to

$$\Gamma_0 = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k}, \quad \theta_k \sim H$$

where $\{\pi_k, \theta_k\}_{k=1}^{\infty}$ is the global set of topics. The idea is to consider $\{\pi_k, \theta_k\}_{k=1}^{\infty}$ as a global topic pool, and each document just selects its own topics from this pool. In this way, the probability of sharing topics between different documents will not be 0. This chapter uses a thinned Gamma process to realize this idea. Its definition is as follow,

Definition 7 (Thinned Gamma Process (Foti et al., 2013)) *Suppose there are countably infinite points $\{(\pi_k, \theta_k)\}_{k=1}^{\infty}$ from a Gamma process $\Gamma \sim GaP(\alpha, H)$. This chapter uses a thinned Gamma process to realize this idea. Its definition is* *as follow,*

$$\Gamma' = \sum_{k=1}^{\infty} \pi_k r_k \delta_{\theta_k} \tag{5.5}$$

is still a Gamma process, which is proofed by (Foti et al., 2013). The $\{r_k\}$ *can be seen as the indicators for the reservation of the point of original/global Gamma process, so* Γ' *is called Thinned Gamma Process.*

Each r_k is assigned a Bernoulli prior $p(r_k = 1) = q_k$, where $q_k \in [0, 1]$ is the subsampling probability of keeping topic k. Apparently, different realizations of $\{r_k\}$ will lead to different thinned Gamma processes. For each document, a thinned Gamma process Γ_d is generated with Γ_0 as the global process

$$\Gamma_d = \sum_{k=1}^{\infty} \pi_k r_{d,k} \delta_{\theta_k} \tag{5.6}$$

where $\{r_{d,k}\}_{k=1}^{\infty}$ is a set of indicators of document d on the corresponding components. These $\{r_{d,k}\}_{k=1}^{\infty}$ are independent identical distributed random variables with Bernoulli distributions

$$r_{d,k} \sim Bernoulli(q_{d,k}) \tag{5.7}$$

where $q_{d,k}$ denotes the probability of the Gamma process Γ_d of document d with component k. Until now, Challenge 2 is handled.

Challenge 3 *How to make two linked documents have similar topics?*

Two linked documents in a document network normally have similar topics. For example, the linked two academic papers through a citation are normally with some common researches, and two linked webpages through a hyperlink normally report similar news. Therefore, it needs to control the sharing strategy of documents on the infinite topics in order to make two linked documents have similar topics.

Since each document has been assigned a thinned Gamma process, the idea is to make thinned Gamma processes dependent with each other according to the document network structure. Each thinned Gamma process is subsampled from the global Gamma process Γ_0 according to indicators $\{r_k\}$. Therefore, the dependence between the different realizations of $\{r_k\}$ will also lead to dependence of the thinned Gamma processes.

In order to obtain the dependent $\{r_k\}$ between documents, a Subsampling Markov Random Field (MRF) is defined to constrain the q_k^d of all documents,

Definition 8 *[Subsampling Markov Random Field] The subsampling probabilities of all the documents on a component/topic in the global Gamma process have the following constraint*

$$\begin{aligned} p(\boldsymbol{q_k}) &= \prod_{C \in \wp(Network)} \psi(C) \\ &= \frac{1}{Z(\boldsymbol{q_k})} \exp\left(- \sum_{<d_i,d_j> \in C} \|q_{d_i,k} - q_{d_j,k}\|^2 \right) \end{aligned} \tag{5.8}$$

where $\boldsymbol{q_k} = \{q_{d,k}\}_{d=1}^{D}$*;* $Network$ *is the document network;* $\wp(Network)$ *is the clique set of* $Network$*;* C *is one clique;* $\psi(C)$ *is the energy function of MRF;* $< d_i, d_j > \in C$ *denotes there is link between* d_i *and* d_j *and this link is within clique* C*; and* $Z(\boldsymbol{q_k})$ *is the normalization part and also called partition function.*

Note that the energy function $\exp\left(-\sum_{<d_i,d_j>\in C} \|q_{d_i,k} - q_{d_j,k}\|^2\right)$ in Definition 8 is designed to constrain the distance between q of different documents on topic k. The more closely two documents are posited in the network, the more close their q on topic k. Through this subsampling MRF constraint, the marginal distribution of each subsampling probability q_k will depend on the values of its neighbors. Therefore, the q_k of linked documents will be similar, which ensures the Challenge 3 is handled.

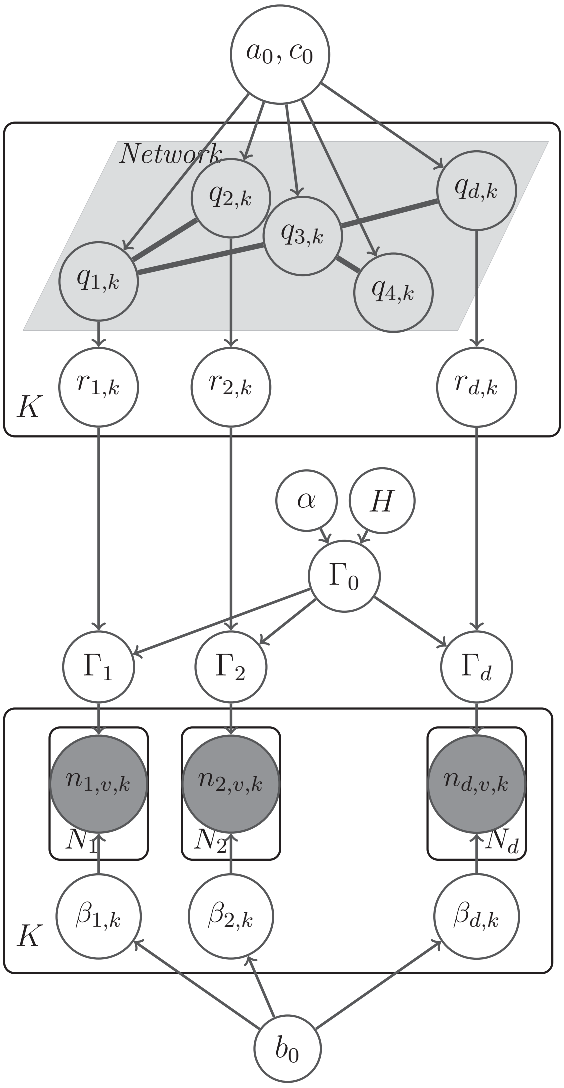


Figure 5.3: Graphical representation for the Nonparametric Relational Topic (NRT) Model by dependent thinned Gamma Processes and Markov Random Field (MRF). The generative procedures for the document 3 and 4 are omitted to make the figure more concise.

To sum up, the proposed Nonparametric Relational Topic (NRT) Model is graphically illustrated in Fig. 5.3 and its generative procedure is

$$\Gamma_0 \sim GaP(\alpha, H)$$

$$
\begin{aligned}
or \quad \Gamma_0 &= \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k} \\
p(q_{d,k}) &\propto Beta(q_{d,k}; a_0, c_0) \cdot \exp\left(- \sum_{\substack{< d, d_i > \in C \\ C \in \wp(Network)}} \|q_{d,k} - q_{d_i,k}\|^2 \right) \\
r_{d,k} &\sim Bernoulli(q_{d,k}) \\
\Gamma_d &= \sum_{k=1}^{\infty} r_{d,k} \pi_k \delta_{\theta_k}
\end{aligned}
$$

With the $\{\Gamma_d\}_{d=1}^{D}$ for all the documents in hand, the generative procedure of the documents is as follow

$$
\begin{aligned}
\beta_{d,k} &\sim Gamma(b_0, 1) \\
n_{d,v,k} | n_{d,v} &\sim Poisson(\theta_{k,v} r_{d,k} \pi_k \beta_{d,k}) \\
v &\in [1, V] \\
n_{d,v} = \sum_{k=1}^{\infty} n_{d,v,k} &\sim Poisson(\sum_{k=1}^{\infty} \theta_{k,v} r_{d,k} \pi_k \beta_{d,k}) \\
\theta_k &\sim H
\end{aligned}
$$

where $n_{d,v}$ is the number of word v in document d (same word may appear several times in a document), $n_{d,v,k}$ is the number of word v in document d assigned to topic k, and $\beta_{d,k}$ is a parameter. Considering the relationship between the Poisson distribution and the Multinomial distribution, the likelihood part is equal to

$$
\begin{aligned}
z_{d,v,m} &\sim Discrete\left(\frac{\theta_{k,v} r_{d,k} \pi_k \beta_{d,k}}{\sum_k \theta_{k,v} r_{d,k} \pi_k \beta_{d,k}} \right) \\
m &\in [1, n_{d,v}]
\end{aligned}
$$

$$n_{d,v,k} = \sum_m \delta_k(z_{d,v,m})$$

$$\theta \sim H$$

This form is more convenient for the slice sampling design for the model which will be explained in the following Section. a_0, b_0, c_0, α are model parameters. H is a base measure for the global Gamma process, and it is set as a Dirichlet distribution parameterized by η. Note that the $\boldsymbol{q}$ are not only with Beta distribution prior but also with a MRF constraint at the same time.

5.4 Model inference

The inference of the proposed (NRT) model is to compute the posterior distribution of latent variables given data (i.e., document network)

$$p(K, \pi, q, r, \theta, \beta | \{n_{d,v}\}_{d\in[1,D],v\in[1,V]}, Network)$$

It is apparently that this posterior distribution is a high-dimensional and multivariable distribution which analytical form is extremely hard to obtain. Therefore, this chapter first uses Gibbs sampling method to get samples of this posterior distribution with a truncation (define a relatively large topic number), which is a commonly-adopted strategy in the Bayesian nonparametric learning area (Gershman and Blei, 2012) in Section 5.4.1. Furthermore, an exact sampling method without the truncation requirement is also developed based on slice sampling technique (Neal, 2003b) in Section 5.4.2.

5.4.1 Gibbs sampling

It is difficult to perform posterior inference under infinite mixtures, and a common work-around solution in Bayesian nonparametric learning is to use a truncation method. This method is widely accepted, which uses a relatively big $K^\dagger$ as the (potential) maximum number of topics. As required by the Gibbs sampling framework, all the conditional distributions for the latent variables of the model are list in the following.

Sampling $q_{d,k}$**.** Since there are additional constraints for the variables q, they do not have a closed-formed posterior distribution.

If $r_{d,k} = 1$,

$$p(q_{d,k}|\cdots) \propto q_{d,k}^{a_0+1-1}(1-q_{d,k})^{c_0-1} \cdot \exp\left(- \sum_{\substack{< d, d_i > \in C \\ C \in \wp(Network)}} \|q_{d,k} - q_{d_i,k}\|^2\right) \tag{5.9}$$

If $r_{d,k} = 0$,

$$p(q_{d,k}|\cdots) \propto q_{d,k}^{a_0-1}(1-q_{d,k})^{c_0+1-1} \cdot \exp\left(- \sum_{\substack{< d, d_i > \in C \\ C \in \wp(Network)}} \|q_{d,k} - q_{d_i,k}\|^2\right) \tag{5.10}$$

Given this conditional distribution of $q_{d,k}$, the efficient $A*$ *sampling* (Maddison et al., 2014) that is developed recently can be used, because the conditional distribution can be decomposed into two parts: $q_{d,k}^{a_0-1}(1-q_{d,k})^{c_0+1-1}$ and exponential part. The first part is easily sampled using a beta distribution (proposal distribution), and the second part is a bounded function.

Sampling $r_{d,k}$**.**

1. $\forall j, r_{d,j} = 0 \rightarrow r_{d,k} = 1$

2. $\exists v, n_{d,v,k} > 0 \rightarrow r_{d,k} = 1$

3. $\forall v, n_{d,v,k} = 0$

   (a) if $\forall v, u_{d,v,k} = 0$,

$$p(r_{d,k} = 1) \propto q_{d,k} \prod_{n} Poisson(0; \theta_{k,v}\pi_k\beta_{d,k}) \tag{5.11}$$

   (b) if $\forall v, u_{d,v,k} = 0$,

$$p^{(1)}(r_{d,k} = 0) \propto (1 - q_{d,k}) \prod_{n} Poisson(0; \theta_{k,v}\pi_k\beta_{d,k}) \tag{5.12}$$

   (c) if $\exists v, u_{d,v,k} > 0$,

$$p^{(2)}(r_{d,k} = 0) \propto (1 - q_{d,k}) \cdot \left(1 - \prod_{v} Poisson\left(0; \theta_{k,v}\pi_k\beta_{d,k}\right)\right) \tag{5.13}$$

Accordingly,

$$p(r_{d,k} = 1|\cdots) \propto \frac{p(r_{d,k} = 1)}{p(r_{d,k} = 1) + p^{(1)}(r_{d,k} = 0) + p^{(2)}(r_{d,k} = 0)} \tag{5.14}$$

Sampling $\beta_{d,k}$. $\beta_{d,k}$ is a model parameter with a Gamma prior and due to the conjugate between the Gamma and Poisson distribution,

$$p(\beta_{d,k}|\cdots) \propto Gamma(n_{d,\cdot,k} + b_0, \frac{1}{r_{d,k}\pi_k + 1}) \tag{5.15}$$

where $n_{d,\cdot,k} = \sum_v n_{d,v,k}$ is the number of words assigned to topic k in document d.

Sampling θ_k. In the proposed model, H is set as a probability (Dirichlet) distribution parameterized by η, so the posterior is as follows

$$p(\theta_k|\cdots) \propto Dirichlet(\eta + n_{\cdot,1,k}, \ldots, \eta + n_{\cdot,V,k}) \tag{5.16}$$

where $n_{\cdot,v,k} = \sum_d n_{d,v,k}$ is the number of word v assigned to topic k in all the documents.

Sampling $n_{d,v,k}$. (truncated version) Here, it needs to sample the $n_{d,v,1}$, ..., $n_{d,v,K^\dagger}$ together due to the known $n_{d,v} = \sum_{k=1}^{K^\dagger} n_{d,v,k}$ according to Multinomial distribution

$$p(n_{d,v,1}, \ldots, n_{d,v,K^\dagger}|\cdots) \propto Mult(n_{d,v}; \xi_{d,v,1}, \ldots, \xi_{d,v,K^\dagger}) \tag{5.17}$$

where

$$\xi_{d,v,k} = \frac{\theta_{k,v} r_{d,k} \pi_k \beta_{d,k}}{\sum_k^{K^\dagger} \theta_{k,v} r_{d,k} \pi_k \beta_{d,k}}. \tag{5.18}$$

Sampling π_k. (truncated version) Although is from a Gamma process, it can be seen with a Gamma distribution prior given a truncation level $K^\dagger$, so the following posterior can be sampled

$$p(\pi_k|\cdots) \propto Gamma(1/K^\dagger + n_{\cdot,\cdot,k}, \frac{1}{\beta_{\cdot,k} + 1}) \tag{5.19}$$

where $n_{\cdot,\cdot,k} = \sum_d \sum_v n_{d,v,k}$ is the total number of words assign to topic k and $\beta_{\cdot,k} = \sum_d \beta_{d,k}$. Note that the truncation version of the model is not equal to a probability distribution-based model (Fox et al., 2011*b*). Under this truncation, there is only limited of topics will be used by documents and most of others will be unused. This truncation can be seen as an approximation of the NRT.

The whole sampling algorithm is summarized in Algorithm 5.1. It is interesting that the q_k of different d are independent of each other given other variables. So the update of q_k of different d can be implemented in a parallel fashion.

Algorithm 5.1: Truncated Version of Gibbs Sampling for NRT

Input: $Network$ and $\boldsymbol{n_{d,v}}$
Output: K, $\{\theta_k\}_{k=1}^K$, $\{\pi_k^d\}_{k=1}^K$
initialization;
$it = 1$; **while** $it \leq max_{it}$ **do**
  for *each topic* k **do**
    for *each document* d **do**
      for *each word* v *of document* d **do**
        Update $n_{d,v,k}$ by Eq. (5.17) ;
      Update $q_{d,k}$ by Eq. (5.9) and (5.10) ;
      Update $r_{d,k}$ by Eq. (5.14) ;
      Update $\beta_{d,k}$ by Eq. (5.15) ;
    Update θ_k by Eq. (6.65) ;
    Update π_k by Eq. (5.19);
  $it+$;

5.4.2 Slice sampling

Although the truncated method are commonly accepted in the literature, maintaining a large number of components and their parameters is time and space consuming. An elegant idea (call slice sampling (Neal, 2003*b*)) to resolve this problem is to introducing additional variables to adaptively truncate/select the infinite components.

Sampling $n_{d,v,k}$ (slice sampling version) In order to do slice sampling, the following auxilary/slice variable is introduced

$$u_{d,v,m} = Uniform(0, \zeta_k), m \in [1, n_{d,v}] \tag{5.20}$$

where $Uniform(0, \zeta_k)$ is a Uniform distribution on $[0, \zeta_k]$ and ζ_k is a fixed positive decreasing sequence $lim_{k\to\infty}\zeta_k = 0$. With the help of slice variable $u_{d,v,m}$, sample the $z_{d,v,m}$ within a finite scope as

$$
\begin{aligned}
p(z_{d,v,m} = k|\cdots) &\propto \xi_{d,v,k} \cdot \frac{\Pi(u_{d,v,m} \le \zeta_k)}{\zeta_k} \\
n_{d,v,k} &= \sum_m \delta_k(z_{d,v,m}) \\
m &\in [1, n_{d,v}]
\end{aligned} \tag{5.21}
$$

where $\Pi(u_{d,v,m} \le \zeta_k) = 1$ when $u_{d,v,m} \le \zeta_k$ is satisfied; $\Pi(u_{d,v,m} \le \zeta_k) = 0$ when $u_{d,v,m} \le \zeta_k$ is not satisfied. Note that the possible values of $z_{d,v,m}$ are limited by $\Pi(u_{d,v,m} \le \zeta_k)$ because ζ_k is a fixed positive decreasing sequence.

Sampling π_k (slice sampling version) The construction of Gamma process $\Gamma_0 \sim GaP(\alpha, H)$ (Roychowdhury and Kulis, 2015) is

$$
\Gamma_0 = \sum_{k=1}^{\infty} E_k e^{-T_k} \delta_{\theta_k} \tag{5.22}
$$

where E_k and T_k are two additional auxiliary variables as

$$
E_k \sim Exp(\frac{1}{\alpha}),\ T_k \sim Gamma(\kappa_k, \frac{1}{\alpha}),\ \theta_k \sim H \tag{5.23}
$$

where $Exp(\frac{1}{\alpha})$ denotes an Exponential distribution parameterized by $\frac{1}{\alpha}$. According to (Wang and Carin, 2012; Roychowdhury and Kulis, 2015), all the components/points/topics could be considered as draws from a number (I and I could be infinitely large) of Poisson processes, so each topic is assigned a Poisson process index κ_k and the following property holds

$$
\sum_{k=1}^{\infty} \delta_{\kappa_k}(i) \sim Poisson(\gamma), \quad \gamma = \int_{\Theta} H \tag{5.24}
$$

which means that the number of topics from each Poisson process satisfies a

Poisson distribution parameterized by γ that is the total mass of base measure H of Gamma Process. Note that γ is equal to 1 if the H is set as a probability measure. Finally, According to the construction in Eq. (5.22), the prior of π_k is

$$\pi_k = E_k e^{-T_k} \sim Exp(\frac{1}{\alpha}) \cdot Gamma(\kappa_k, \frac{1}{\alpha}) \tag{5.25}$$

and the posterior is

$$\pi_k = (E_k, T_k) \sim Poisson(data|E_k e^{-T_k}) Exp(E_k|\frac{1}{\alpha}) Gamma(T_k|\kappa_k, \frac{1}{\alpha}) \tag{5.26}$$

This posterior can be sampled by two Gamma distributions as

$$\begin{aligned} E_k|T_k &\sim Gamma(E_k|n_{\cdot,\cdot,k}+1, \frac{1}{\alpha^{-1}+\beta_{\cdot,k}e^{-T_k}}) \\ T_k|E_k &\sim Poisson(n_{\cdot,\cdot,k}|\beta_{\cdot,k}e^{-T_k}E_k) \cdot Gamma(T_k|\kappa_k, \frac{1}{\alpha}) \end{aligned} \tag{5.27}$$

where $n_{\cdot,\cdot,k} = \sum_d \sum_v n_{d,v,k}$ and $\beta_{\cdot,k} = \sum_d \beta_{d,k}$.

The conditional distribution for the indicator κ_k is

$$p(\kappa_k = i|\cdots) \propto p(T_k|\kappa_k = i) \cdot p(\kappa_k = i|\{\kappa_l\}_{l=1}^{k-1}) \tag{5.28}$$

The second part on the right hand side of Eq. (5.28) is

$$\begin{aligned} & p(\kappa_k = i|\cdots) \\ = & \begin{cases} 0 & \text{if } i < \kappa_{k-1} \\ \dfrac{1-F(I_{i-1}|\gamma)}{1-F(I_{i-1}-1|\gamma)} & \text{if } i = \kappa_{k-1} \\ \dfrac{(F(I_{i-1}|\gamma)-F(I_{i-1}-1|\gamma))}{1-F(I_{i-1}-1|\gamma)} \left(1-f(0|\gamma)\right) f(0|\gamma)^{h-1} & \text{if } i = \kappa_{k-1}+h \end{cases} \end{aligned} \tag{5.29}$$

where h is an integer denotes the distance between κ_k with κ_{k-1}; I_i is the number

Algorithm 5.2: Slice Version of Gibbs Sampling for NRT

```
Input: Network and n_{d,v}
Output: K, {θ_k}_{k=1}^K, {π_k^d}_{k=1}^K
initialization;
it = 1;
while it ≤ max_it do
    for each topic k do
        for each document d do
            for each word v of document d do
                Sample slice variable n_{d,v,k} by Eq. (5.20) ;
                Update n_{d,v,k} by Eq. (5.21) ;
            Update q_{d,k} by Eq. (5.9) or (5.10) ;
            Update r_{d,k} by Eq. (5.14) ;
            Update β_{d,k} by Eq. (5.15) ;
        Update θ_k by Eq. (6.65) ;
        Update π_k by Eq. (5.27);
        Update κ_k by Eq. (5.28);
    it++;
```

of items in i-th Poisson process and $I_i \sim Poisson(\gamma)$; $F(\cdot|\gamma)$ and $f(\cdot|\gamma)$ are the cumulative distribution function and probability density function of Poisson distribution parameterized by γ.

Note that the $u_{d,v,m}$, κ_k, E_k and T_k are introduced additional variables. They are not in the original model, and their appearances are only for the sampling without the help of the truncation level. The whole slice sampling algorithm is summarized in Algorithm 5.2.

5.5 Experiments

This section evaluates the effectiveness of the proposed NRT model in learning the hidden topics from document networks. First, two evaluation metrics are designed for the quantification of the effectiveness and comparisons. Then, a series of experiments on the synthetic datasets to demonstrate the model's ability of topics and topic number learning and some properties. Finally, it shows the usefulness of the proposed model using two real-world datasets.

5.5.1 Evaluation metrics

Since NRT builds on two parts of knowledge (i.e., the network structure and document content) from a document network data, it is possible to make predictions for one of them based on the other (Chen, Zhu, Xia and Zhang, 2015; Chang and Blei, 2009). In order to quantitatively compare the NRT with other models, two evaluation metrics are designed: *link prediction* and *document prediction*.

The *link prediction* is to predict the links between test documents with training documents using learned topics. The assumption is that there will be a link between two documents if they have similar topics. For two given test documents d_i and d_j, the probability of a link between d_i and d_j is

$$p^l(c_{d_i,d_j} = 1) \propto \cos(\varpi_{d_i}, \varpi_{d_j}) \tag{5.30}$$

where $\cos(\cdot,\cdot)$ is the cosine similarity measure between two vectors and ϖ_{d_i} is a vector and represents the interest of test document d_i on topics which can be evaluated by

$$\varpi_{d_i,k} = \sum_{v=1}^{V} n_{d_i,v} \cdot \theta_{k,v} \tag{5.31}$$

where $n_{d_i,v}$ denotes the number of word v in document d_i, θ_k is the learned topic from documents, and ϖ_{d_i} is the normalized $< \varpi_{d_i,1}, \cdots, \varpi_{d_i,K} >$ where K is the learned topic number. The above Eq. (5.30) accounts for one test link and sum over the probabilities of all test links in the experiments.

The *document prediction* is to evaluate the likelihood of a test document with known network structure with training documents using learned topics.

The assumption is that the likelihood of a test document with similar topic interests with its linked training documents is large. For a given test document d_i with its linked training documents Ω_d, the likelihood of this document can be evaluated by

$$p^d(d_i) \propto \sum_{v=1}^{V} n_{d_i,v} \sum_{k=1}^{K} \theta_{k,v} \cdot \widetilde{\pi}_{d_i,k} \tag{5.32}$$

where $\widetilde{\pi}_{d_i,k}$ is the average of topic interests of neighbors training documents and can be evaluated by

$$\widetilde{\pi}_{d_i,k} = \frac{\sum_{d' \in \Omega_d} \pi_k \cdot r_{d',k}}{|\Omega_{d_i}|} \tag{5.33}$$

where Ω_{d_i} is the neighbor(i.e., training documents) set of test document d_i, $|\Omega_{d_i}|$ is the number of documents in Ω_{d_i}, and $\{\pi_k, r_{d,k}\}$ are from the NRT model. The above Eq. (5.32) accounts for one test document and sum over the probabilities of all test documents in the experiments.

Note that the probability of the test links and documents is used here rather than the exact predicted number of links or documents because the proposed model is kind of generative model same with other similar works (Chen, Zhu, Xia and Zhang, 2015; Chang and Blei, 2009). So the probability evaluations are more accurate and approximate for the proposed model and comparisons.

5.5.2 Experiments on synthetic data

Synthetic data is generated to explore the NRT's ability to learn the hidden topics and infer the number of hidden topics from the document network.

(1) Synthetic data generation

At first, choose a set of ground truth numbers symbolised by K, D and V that refer to the number of topics, documents and (different) keywords, respectively. Then, K global topics are generated through a V-dimensional Dirichlet distri-

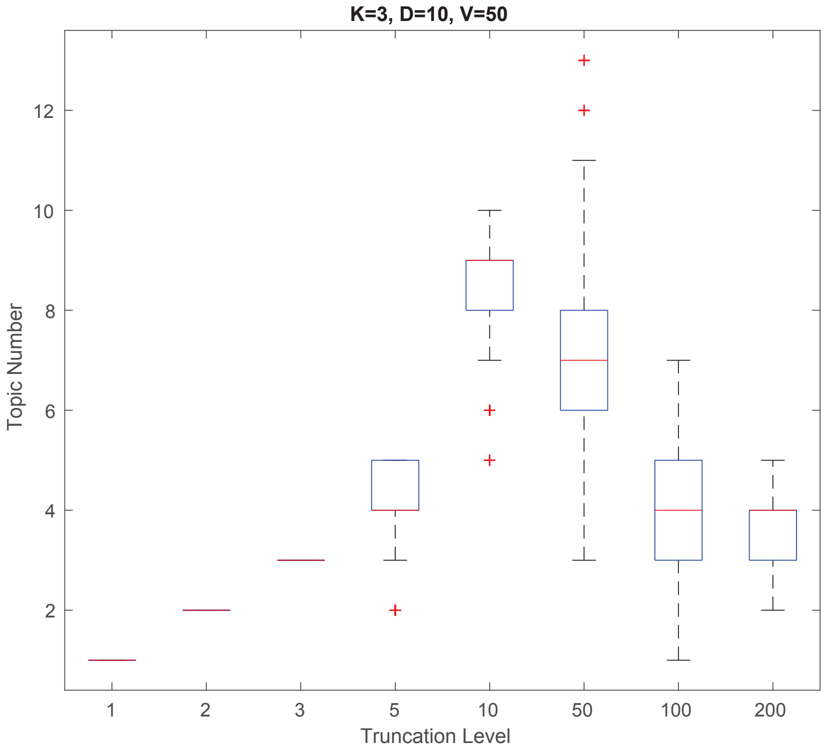


Figure 5.4: The boxplot of the learned topic numbers by truncated inference method given different truncation levels.

bution parameterized by $\{\alpha_1, \ldots, \alpha_V\}$ where $\alpha_i = 1\ \forall i$. Next, generate the document interests on these topics through a K-dimensional Dirichlet distribution parameterized by $\beta_1, \ldots, \beta_K\}\ \forall \beta_i = 1$. With topics and the document interests on these topics in hand, generate each document d as follows: 1) N_d is uniformly chosen to be a number between $\frac{N}{2}$ and N where N is set as the maximum number of words in a document; 2) Repeat the following operations N_d times: a topic index is drawn from the document's topic interest and then draw a word from the selected topic. Finally, obtain a $D \times V$ matrix with rows as documents and columns as words, and each entry of this matrix $n_{d,v}$ denotes the frequency a particular word v in a particular document d. The next step

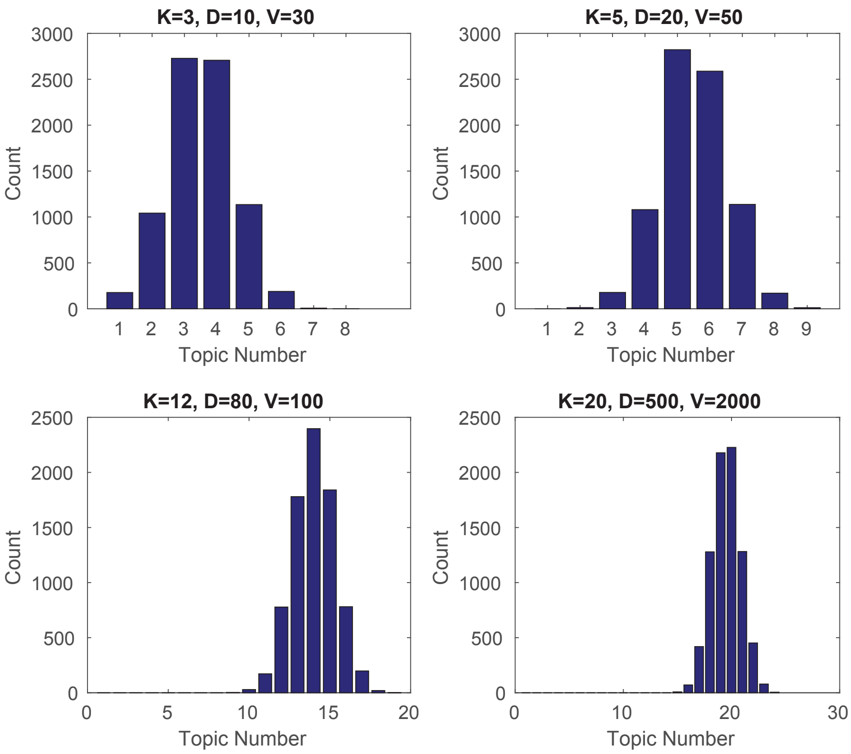


Figure 5.5: Learned topic number distribution from NRT with synthetic datasets under different settings. Normally, the expectation of this distribution will be regarded as the learn topic number of a document network.

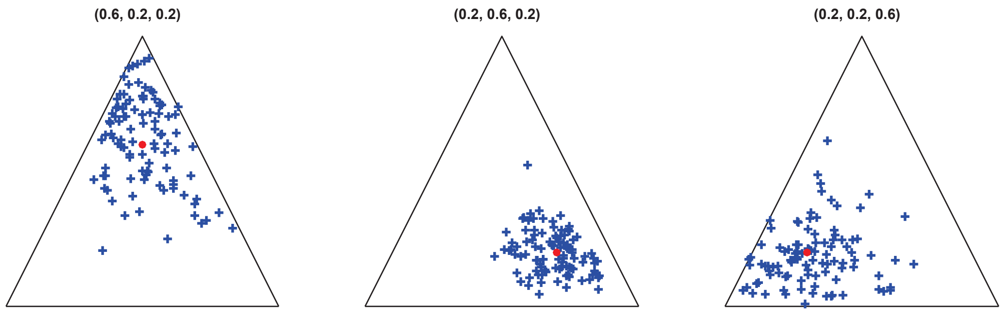


Figure 5.6: The illustration of topics learning results. Three red/circle nodes denote three benchmark topics that are also given at the top of each subfigure; the blue/cross nodes denote learned topics from NRT.

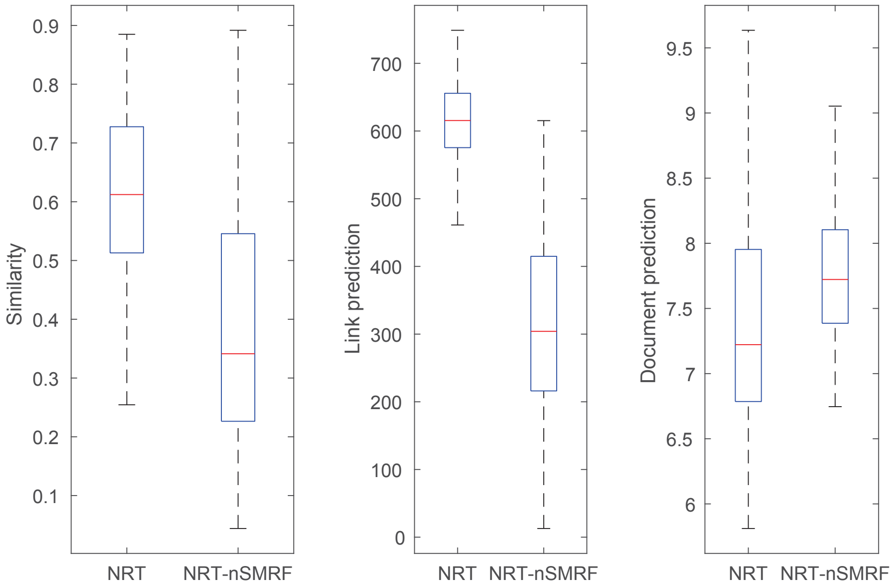


Figure 5.7: Effectiveness of SMRF in NRT. The first subfigure is for the comparision between average similarity between topic interests of all test linked document pairs from both NRT and NRT without SMRF; The second and third subfigures are for comparison on Link and Document prediction.

is to generate the relations between documents. For each pair of documents, compute the inner product between their topic interests. In order to sparsify these relationships, only retain the ones where their inner products are greater than 0.2.

(2) Influence of truncation level

There are two inference methods proposed in this chapter: one is truncation version and the other is slice version. For the truncation version in Algorithm 5.1, a truncation level needs to be given in advance. In order to show the influence of this parameter, different truncation levels (i.e., $K^{\dagger} \in \{1, 2, 3, 5, 10, 50, 100, 200\}$) and a generated dataset using the procedure in Section 5.5.2.1 with the setting (i.e., $K = 3$, $D = 10$ and $V = 50$) are fed into Algorithm 5.1. For each run, it takes 10,000 iterations with 2,000 burn-in samples. The final 800 samples are obtained by keeping each sample every 10 samples considering the dependency between the consecutive samples from the Algorithm. The results are plotted in Fig. 5.4 which shows not only the topic number means from eight truncation levels but also the some basic statistics of 800 samples at each truncation level. It can be seen that the topic number dose not exceed the truncation levels when they are smaller than the real one (i.e., 3 for this dataset). When the truncation level is larger than 3, there will be a fluctuation of the learned topic numbers but the learned topic number will still not exceed the truncation level, so the fluctuation is small when the truncation level is not very large (such as 5 in the Fig. 5.4). As the increasing of the truncation level, the approximation of the truncation version distribution is more accurate, so the learned topic number will be closer to the real one and the variance is smaller.

(3) Topics learning

One ability of NRT model is to discover the hidden topics from a document network. This subsection aims to show this ability. At first, generate a synthetic dataset using the revised procedure in Section 5.5.2.1 with a setting (i.e., $K = 3$, $D = 10$, $V = 3$) and the topics are predefined as benchmarks rather than randomly sampled ones. The three topics are $(0.6, 0.2, 0.2)$, $(0.2, 0.6, 0.2)$, and $(0.2, 0.2, 0.6)$, which correspond to three points in the 3-dimensional simplex. After running NRT model, keep 100 samples with 3 topics. In each sample, there are three learned topics which are linked to the benchmark topics according to the similarity measure, and choose the *best linking status* as the final one for each sample. The *best linking status* means the the total similarity between each pair of topics reaches maximum. For example, there are three learned topics in a sample: $(0.25, 0.5, 0.25)$, $(0.5, 0.25, 0.25)$, and $(0.25, 0.25, 0.5)$. It should link the first learned topic to $(0.2, 0.6, 0.2)$, the second learned topic to $(0.6, 0.2, 0.2)$, and the third learned topic to $(0.2, 0.2, 0.6)$. In Fig. 5.6, three red/circle nodes denote three benchmark topics and the blue/cross ones are from samples. It can be seen from this figure that the samples from NRT centers on the benchmark topics with a certain variance, which shows the effectiveness of NRT on the topics learning.

(4) Topic number learning

Another ability of NRT model is to discover the hidden topics without the requirement of the predefined topic number. In order to show this ability, the synthetic data generation procedure in Section 5.5.2.1 is used with different settings: $K = 3$, $D = 10$, $V = 30$; $K = 5$, $D = 20$, $V = 50$; $K = 12$, $D = 80$, $V = 100$; $K = 20$, $D = 500$, $V = 2000$. For each setting, run the NRT model with 10,000 iterations with first 2,000 samples as burn-in stage. In Fig. 5.5, he topic numbers in the remaining 8,000 samples from NRT model on four synthetic datasets are plotted. From this figure, the conclusion can be drawn that NRT has the ability to learn out the topic number from a document network to some extent.

(5) Effectiveness of SMRF

SMRF in Definition 8 is used to add the network structure into the model. In order to evaluate the performance of this SMRF, this section compares NRT with SMRF and NRT without SMRF using generated dataset by Section 5.5.2.1 with setting: $K = 10$, $D = 30$, $V = 200$. Among all the documents, 23 documents are considered as the training documents with 44 links, and 7 documents are reserved as the test documents with 10 links. After the mixing of sampling (10,000 iterations with 2,000 burn-in samples), 100 samples with 80 as the interval are kept. At first, the average similarity between topic interests of all test linked document pairs is evaluated. The assumption is that the more similar topic interests of two linked documents, the learned topics are more reasonable because the test links are generated using through the topic interest similarity. The result is plotted in the first subfigure of Fig. 5.7, which shows that NRT with SMRF constraint could recover the test links better. Next, they are compared using the metrics proposed in Section 5.5.1. The results are shown in the second and third subfigures of Fig. 5.7. It can be seen from these figures that SMRF helps NRT obtain better performance on the link prediction task since it considers the training document links, and slightly reduces the performance on the document prediction task.

5.5.3 Experiments on real-world data

(1) Datasets and setup

The real-world document network datasets[2] used in this study are:

- **Cora Dataset** It consists of 2708 scientific publications with their citation relations. The citation network consists of 5429 links. The dictionary consists of 1433 unique words.

- **Citeseer Dataset** The CiteSeer dataset consists of 3312 scientific publications. The citation network consists of 4732 links. The dictionary consists of 3703 unique words (Sen et al., 2008).

For each dataset, 5-fold cross validation is used to evaluate the performance of the proposed model comparing with Relational Topic Model (RTM)(Chang

[2] http://linqs.cs.umd.edu/projects/projects/lbc/

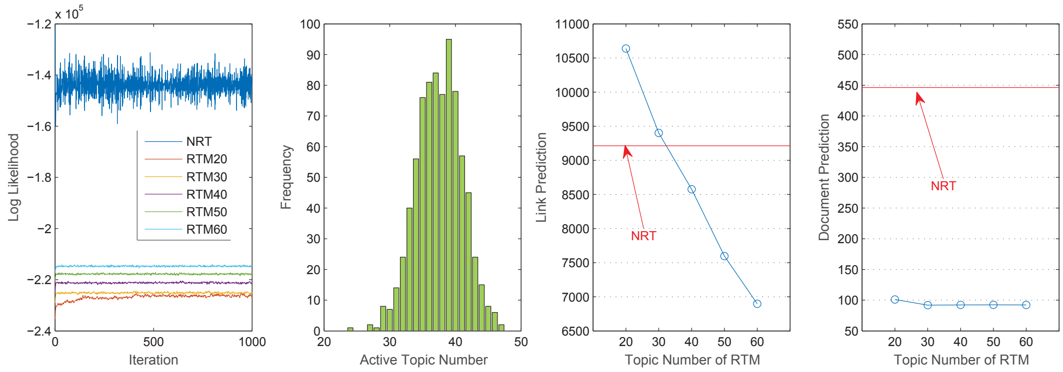


Figure 5.8: Results of NRT and RTM under different setting (K={20, 30, 40, 50, 60}) on a first 5-fold of cora dataset.

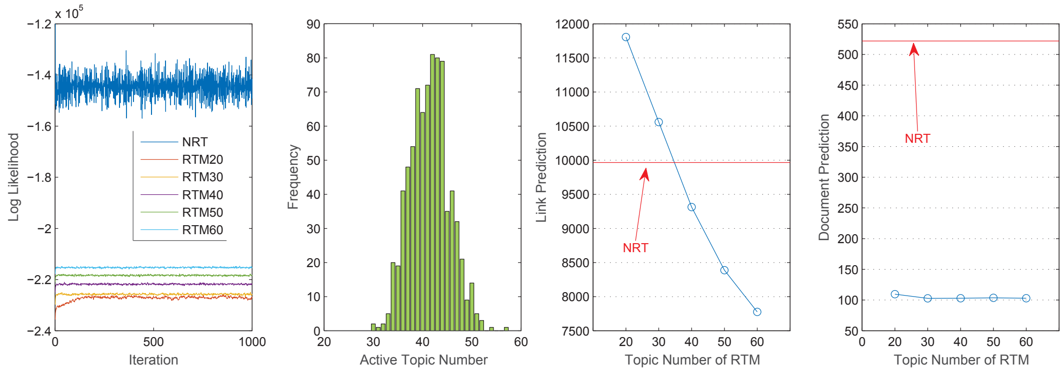


Figure 5.9: Results of NRT and RTM under different settings (K={20, 30, 40, 50, 60}) on a second 5-fold of cora dataset.

et al., 2010). The whole dataset is equally split into five parts. At each stage, documents in one part are chosen for testing while the rest of the four parts are used for training. The implementation of RTM is from *A Fast And Scalable Topic-Modeling Toolbox*[3] for comparison.

(2) Results and discussions

The results are shown in Fig. 5.8, 5.9, 5.10, 5.11, 5.12, 5.13, 5.14, 5.15, 5.16 and 5.17 in which NRT with RTM under several settings have been compared.

[3]http://www.ics.uci.edu/ãsuncion/software/fast.htm#rtm

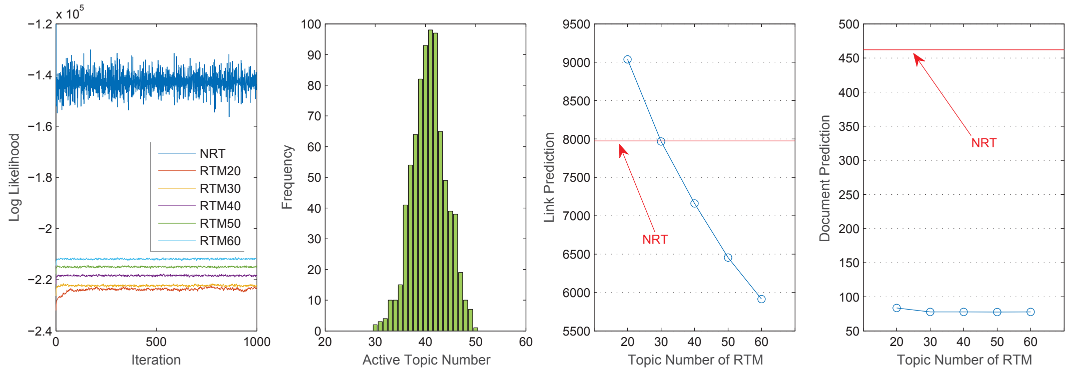


Figure 5.10: Results of NRT and RTM under different settings (K={20, 30, 40, 50, 60}) on a third 5-fold of cora dataset.

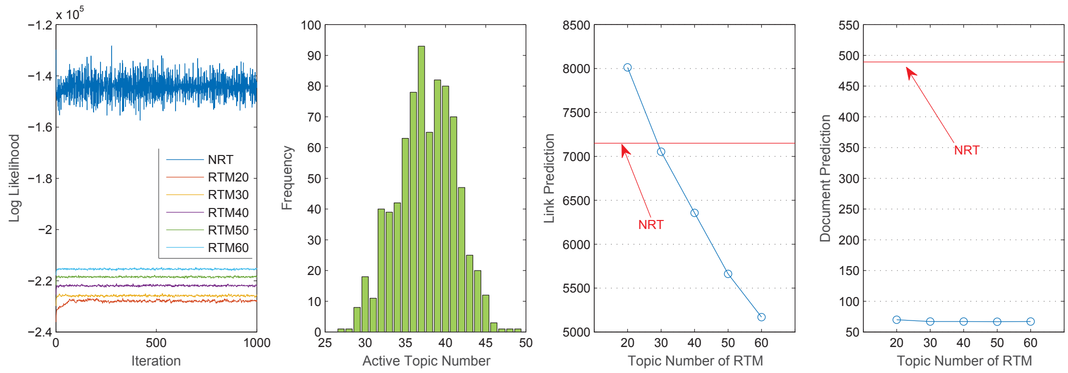


Figure 5.11: Results of NRT and RTM under different settings (K={20, 30, 40, 50, 60}) on a fourth 5-fold of cora dataset.

Since the learned topic number could be different for each fold, the five fold of datasets cannot be simply averaged, so the result on different folds of a dataset are all plotted. The five subfigures at the left hand side (i.e., Fig. 5.8, 5.9, 5.10, 5.11, and 5.12) are for *cora* dataset; the five subfigures at the right hand side (i.e., Fig. 5.13, 5.14, 5.15, 5.16 and 5.17) are for *citeseer* dataset. For clarity, denote RTM with $K = num$ as "RTMnum". For example, RTM20 means RTM with $K = 20$. Since NRT does not need the predefined topic number as an input, it does not impacted by the K and is plotted as a line in figures with K as x-axis. In each figure, there are four subfigures: the first subfigure shows the

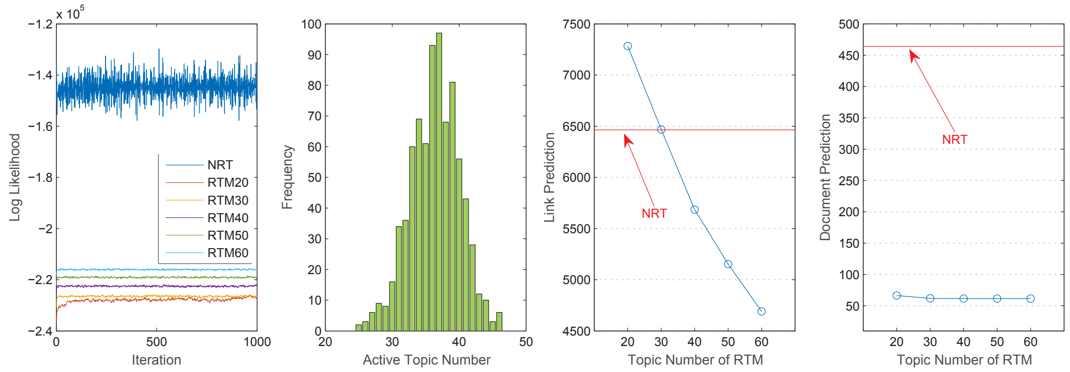


Figure 5.12: Results of NRT and RTM under different settings (K={20, 30, 40, 50, 60}) on a fifth 5-fold of cora dataset.

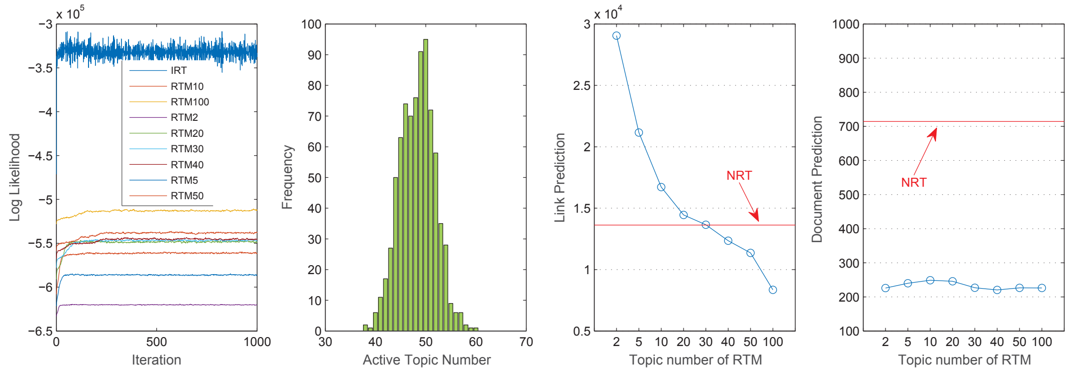


Figure 5.13: Results of NRT and RTM under different settings (K={2, 5, 10, 20, 30, 40, 50, 100}) on a first 5-fold of citeseer dataset.

Log-likelihood along iterations; the second subfigure is the learned distribution of active topic number; the third subfigure is the comparison of NRT and RTM on link prediction task; the fourth subfigure is the comparison of NRT and RTM on document prediction task.

Note that the slice version of NRT in Algorithm 5.2 is used as the implementation of NRT. The reason is that slice version is more efficient than truncated version because the slice version does not need to keep the (relatively) large number of hidden topics in memory (the initial guess for the number of topics is normally set as larger than the number of documents).

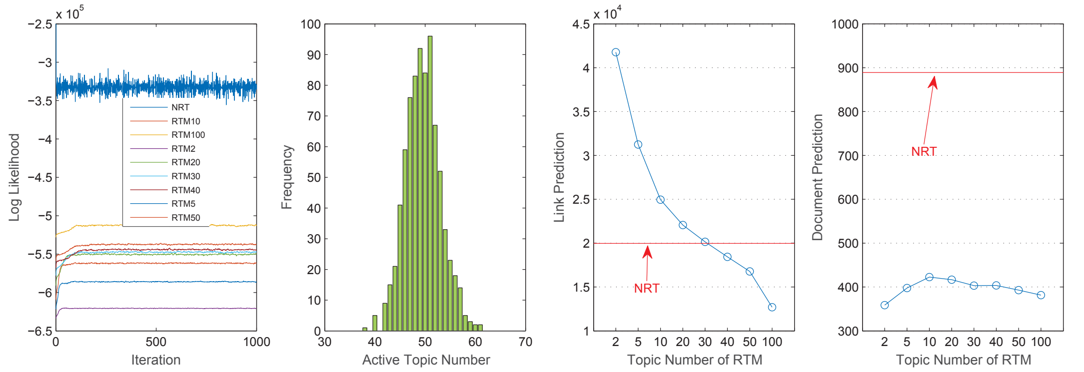


Figure 5.14: Results of NRT and RTM under different settings (K={2, 5, 10, 20, 30, 40, 50, 100}) on a second 5-fold of citeseer dataset.

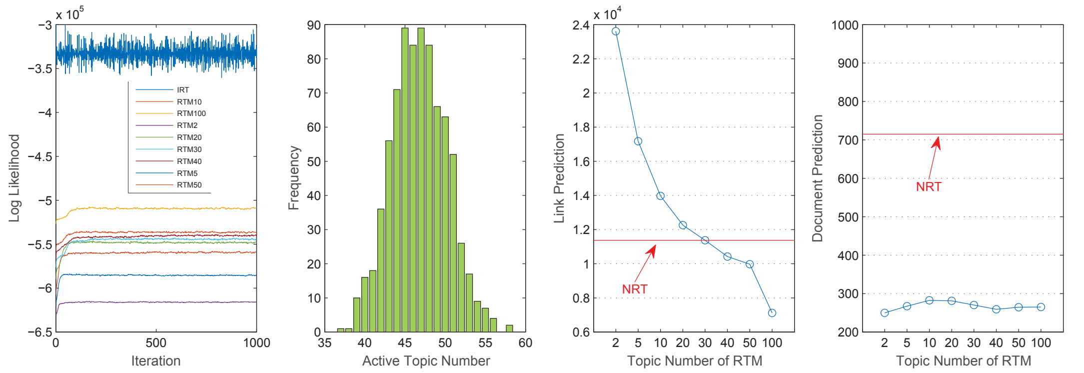


Figure 5.15: Results of NRT and RTM under different settings (K={2, 5, 10, 20, 30, 40, 50, 100}) on a third 5-fold of citeseer dataset.

Note that the proposed algorithm mixed better than some of the K settings in RTM and was generally compatible with the rest. As shown in first part of each subfigure, the likelihood by NRT model was generally larger than the RTM under various settings. It means that the proposed model fits or explains the training data better than the RTM. As for the synthetic case, the distribution of K is also plotted. This section compared the proposed method with RTM in terms of link and document prediction. In terms of document prediction, the proposed algorithm consistently outperformed RTM in every category. In terms of link prediction, NRTąŕs performance was not universally better than

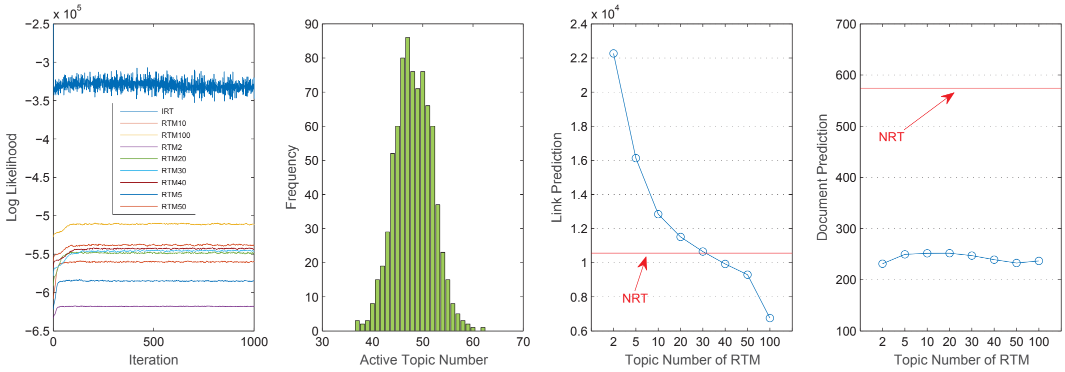


Figure 5.16: Results of NRT and RTM under different settings (K={2, 5, 10, 20, 30, 40, 50, 100}) on a fourth 5-fold of citeseer dataset.

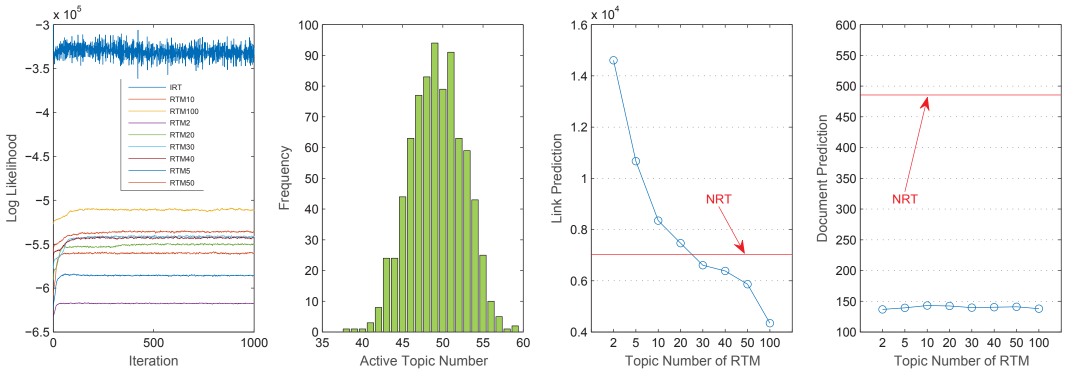


Figure 5.17: Results of NRT and RTM under different settings (K={2, 5, 10, 20, 30, 40, 50, 100}) on a fifth 5-fold of citeseer dataset.

RTM, where it noticed some less accurate results under some RTM settings. It can be seen that there is a trend for the link prediction with respective to the topic number in RTM. This trend comes from the evaluation Eq. (5.30). The RTM with smaller topic number tends to have bigger provability of observed links, which has also been observed in (Chen, Zhu, Xia and Zhang, 2015). At an extreme situation ($K = 1$), the RTM reaches its best performance on link prediction. The problem is how to choose the hidden topic number for RTM. Take *cora* dataset as an example. The candidates of possible topic number are at least within $[1, 2708]$. However, for the proposed NRT model, the active topic number is automatically learned from the data (for *cora* dataset K around 42). Without any prior domain knowledge, this topic number can achieve relatively good results on the link prediction considering its large range $[1, 2708]$. In terms of overall result, it shows that in the absence of an accurate domain knowledge of K value, the NRT algorithm has allowed us achieving better and more robust performance compared with the current state-of-the-art methods.

5.6 Summary

Despite of the success of existing relational topic models in discovering hidden topics from document networks, they are based on the unrealistic assumption, for many real-world applications, that the number of topics can be easily predefined. In order to relax this assumption, this chapter has presented a nonparametric relational topic model. In the proposed model, the stochastic processes are adopted to replace the fixed-dimensional probability distributions used by existing relational topic models which lead to the necessity of pre-defining the number of topics. At the same time, introducing stochastic processes leads to the difficulty with model construction and inference, and this chapter has therefore presented a thinned Gamma process-based model and also presented truncated Gibbs and slice sampling algorithms for the proposed model. Experiments on both the synthetic dataset and the real-world dataset have demonstrated the proposed methodąŕs ability to inference the hidden topics and their number.

Chapter 6

Bayesian Nonparametric Cooperative Hierarchical Structure Models for Multi-label Document Learning

6.1 Introduction

Different from the network structure considered in Chapter 5, this Chapter considers multi-label information of documents. A hierarchical structure has multiple layers, and each layer contains a number of nodes that are linked to the nodes in the higher and lower layers, as illustrated in Fig. 6.1. This structure is a very common and pervasive structure that has been adopted in many different areas. One example of a hierarchical structure is found in text mining. Consider all the papers in a scientific journal (e.g., *Artificial Intelligence*). An *author-paper-word* (Rosen-Zvi et al., 2010) hierarchical structure emerges, given each *author* writes and publishes a number of scientific *papers* in this journal, and each *paper* is composed of several different *words*. There are actually two types of hierarchical structures according to the number of parents for each node: non-cooperative hierarchical structures and cooperative hierarchical structures.

Current renowned Bayesian approaches for the hierarchical structure modeling are mainly based on topic models (Blei et al., 2003; Blei, 2012; Xuan, Lu, Zhang and Luo, 2015) that were originally designed for modeling two-layer hierarchical structure: *document-word*. Beyond the two-layer hierarchical structure, many additional information of data are incorporated into the topic model, so new hierarchical structures have also been successfully modelled, such as: *author-document-word* (Rosen-Zvi et al., 2004; Steyvers et al., 2004; Rosen-Zvi et al., 2010), *emotion-document-word* (Bao et al., 2012), *entry-document-word* (Kim et al., 2012) and *label-document-word* (Zhu et al., 2012). One major issue in existing (parametric) topic model-based hierarchical structure modeling is that the hidden topic number in the defined priors needs to be fixed in advance. An elegant approach to resolve the above issue is *Bayesian nonparametric learning*. Although there are state-of-the-art works in existing Bayesian nonparametric learning could capture non-cooperative hierarchical structures but they fail on the more general one: cooperative hierarchical structures.

This chapter proposes two Bayesian nonparametric models for the cooperative hierarchical structure learning: *Cooperative Hierarchical Dirichlet Process* and *Mixed Gamma-Negative Binomial Process*. More specifically, two operations on the random measures are defined: *Inheritance* which is from the Hierarchical Dirichlet Process; *Cooperation* is an innovation proposed in this chapter to account for the multiple parent nodes in the cooperative hierarchical structure. Although the proposed models elegantly captures cooperative hierarchical structures, it does bring additional challenges to model inference. To resolve this challenge, this chapter introduces a set of additional latent variables which makes

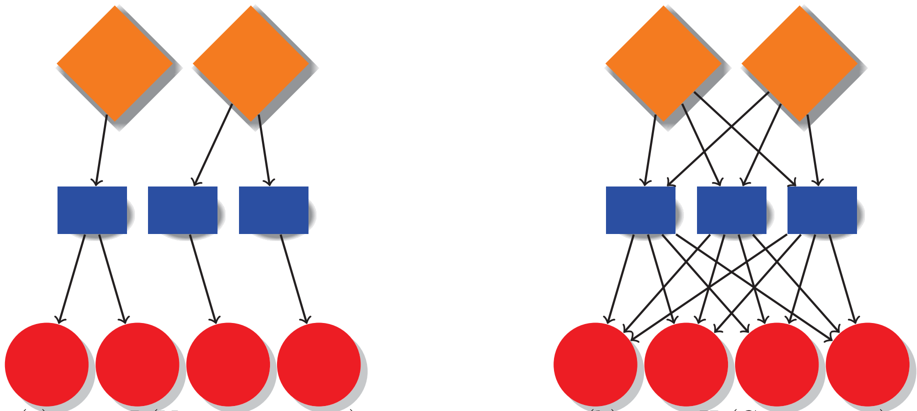

(a) Type-I (Non-cooperative) (b) Type-II (Cooperative)

Figure 6.1: Two types of hierarchical structures

the model both easy to infer as well as scale to many layers. Finally, the expectations of the topic number from two models are theoretically and empirically analyzed. Experiments on two multi-label document learning tasks have shown the efficiency of the proposed models on the cooperative hierarchical structure modeling.

The rest of this chapter is organized as follows. Cooperative hierarchical structure is formally defined in Section 6.2. The Cooperative Hierarchical Dirichlet Process and Mixed Gamma-Negative Binomial Process are presented in Section 6.3 and Section 6.4, respectively. Section 6.5 conducts the comparative experiments on two multi-label document learning tasks. Section 6.6 summarizes this chapter.

6.2 Cooperative hierarchical structure

As discussed in Introduction, there are two types of hierarchical structures according to the number of parents for each node, as illustrated in Fig. 6.1. In Type-I hierarchical structure, as illustrated in Fig. 6.1(a), each node is with one and only one parent node. The nodes in this structure could be seen as assigned to different groups and, in turn, assigned to the higher level groups. In Type-II hierarchical structure, as illustrated in Fig. 6.1(b), each node could have more than one parent node. This structure is named *Cooperative Hierarchical Structure* in this chapter. Apparently, Type-II is more general than Type-I because Type-I could be seen as a special case of Type-II. Note that existing Bayesian nonparametric models (such as, Hierarchical Dirichlet Process (Teh, 2006) and its variances (Dai and Storkey, 2015; Paisley et al., 2015; Canini and Griffiths, 2011)) are all particularly designed after Type-I hierarchical structures but fail to model Type-II Hierarchical Structures. Consider the example of an *author-paper-word* structure. When using the Type-I hierarchical structure to model it, there must be an implication that each *paper* is only written by one *author*. Despite of certain rationality in some situations, the constraint of Type-I hierarchical structure is still too restricted to model many real-world phenomenons (especially for multi-label document learning).

Next, the second type is formally defined:

Definition 9 (Cooperative Hierarchical Structures) *A* cooperative hierarchical structure [CHS]*, as illustrated in Fig. 6.1(b), is composed of nodes assigned to different layers. Each node in the structure could be linked to multiple parent nodes and child nodes.*

A real-world example of the cooperative hierarchical structure is: *author-paper-word* data. There are three-layer nodes in this data: nodes in first layer denote *authors*; nodes in second layer denote *papers*; nodes in third layer denote *words*. If an *author* writes a *paper*, there will be a link between two corresponding nodes; similarly, there will be a link between a *paper* and a *word* if this paper contains this word. The authors here could be seen as labels associated with documents, so *authors* could be replaced by any other kinds of labels. In this chapter, *author-paper-word* is used as example for multi-label documents. Some important notations of this chapter are summarized in Table 6.1.

Table 6.1: Important notations for this chapter

Symbols	Description
Θ	a measurable space
G	a random measure from DP
G_0/G_0^1	global random measure from DP at the first layer
G_a/G^2	a random measure from DP at the second layer
G_d/G^3	a random measure from DP at the third layer
G_i^ℓ	i-th random measure from DP at the ℓ-th layer
N^ℓ	the number of random measures at ℓ-th layer
H	base measure of DP
γ	the parameter of H (when it is Dirichlet distribution)
Ω	a random partition
k	an index of partition/factor/topic/dish
K	the number of partitions/factors/topics/dishes
a	a chef/node at the second layer
A	number of chefs/nodes at the second layer
d	a restaurant
D	number of restaurants/nodes at the third layer
t	a table in a restaurant
T_d	the table number in restaurant d
T_d^a	the table number in restaurant d served by chef a
$T_{a,o}$	the number of tables served by menu option o of chef a
T_k	the number of tables served by dish k
o	a menu option on the personal menu
O_a	the number of menu options on the personal menu of chef a
O_k	the number of menu options with dish name k
V	the number of different words in a corpus
θ_k	k-th partition/factor/topic/dish of DP (one point in Θ)

$\theta_{a,o}$	assigned factor to menu option o of chef a
$\theta_{d,t}$	assigned factor to table t in restaurant d
$\theta_{d,n}/\theta_{d,i}$	assigned factor to data/customer n/i in restaurant d
α	concentration parameter of general DP
α_0	concentration parameter of global DP at first layer
α_a	concentration parameter of DPs at second layer
α_d	concentration parameter of DPs at third layer
ν_k	k-th stick break from beta distribution $Beta(1, \alpha)$
π_k	the stick weight of k-th atom/factor from general DP
$\nu_{0,k}$	k-th stick break from beta distribution $Beta(1, \alpha_0)$
$\pi_{0,k}$	the stick weight of k-th atom/factor from global DP at first layer
$\nu_{a,o}$	o-th stick break from beta distribution $Beta(1, \alpha_a)$
$\pi_{a,k}$	the stick weight of k-th atom/factor from DP at second layer
$\nu_{d,t}$	t-th stick break from beta distribution $Beta(1, \alpha_d)$
$\pi_{d,k}$	the stick weight of k-th atom/factor from DP at third layer
$z_{a,o}$	the assigned index of factor/dish of a node at first layer for menu option/factor o of node a at second layer
$z_{d,t}$	the assigned index of factor/menu option of a node at second layer for table/factor t of node d at third layer
$z_{d,n}$	the assigned index of factor/table of a node at third layer for a data n of d
$\hat{z}_{d,n}$	the assigned index of node at second layer to the data n of d
N_d	the number of data/customers in restaurant d
$N_{d,t}$	the number of data/customers sitting at table t of restaurant d
$N^a_{d,t}$	the number of data/customers sitting at table t of restaurant d served by chef a
$\varpi_{d,a}$	the weight of link between node a and node d
AD	author-document mapping matrix
DN	document-word mapping matrix
A_d	number of authors of document d
Γ_0	a global random measure from a Gamma process
$r_{0,k}$	the global weight of topic k

Γ_d	a random measure from a Gamma process for document d
$r_{d,k}$	the weight of topic k in document d (the interest of d on k)
Γ_a	a random measure from a Gamma process for author a
$r_{a,k}$	the weight of topic k in author a (the interest of a on k)
Γ_a^d	the mixed measure of measures of all authors who write d
$r_{a,k}^d$	the average weight of topic k in all author a who write document d
X	a random measure from a Negative binomial process
n_k	number of words assigned to topic k
X_d	a random measure for document d from a negative binomial process
$n_{d,k}$	number of words assigned to topic k in document d
$n_{a,k}$	number of words assigned to topic k and author a
$n_{d,k}^a$	number of words assigned to topic k and author a in document d

6.3 Cooperative hierarchical Dirichlet processes

The cooperative hierarchical Dirichlet process is built on the existing Bayesian nonparametric prior: hierarchical Dirichlet process (HDP). This section reviews its definitions and constructive representations that have been used to understand and build the proposed CHDP in the following.

Hierarchial Dirichlet Processes (Teh, 2006) is built by piling a DP above another DP through an elegant method which can make factors shared across the hierarchical structure. Its definition is as follow

Definition 10 (Hierarchical Dirichlet Process) *A* Hierarchical Dirichlet Process (HDP) *(Teh, 2006) is a distribution over a set of random probability measures over Θ. The process defines a set of random probability measures $\{G_d\}_{d=1}^D$ and a global random probability measure G_0 as shown in Fig. 6.2(a). The global measure G_0 is distributed as a Dirichlet process parameterized by a concentration* *parameter α and a base (probability) measure H*

$$G_0 \sim DP(\alpha, H) \tag{6.1}$$

Meanwhile, each random measure G_d is conditionally independent with others given G_0, and is also distributed as a Dirichlet process with parameter α_d and a base probability measure G_0

$$G_d \sim DP(\alpha_d, G_0) \tag{6.2}$$

This definition actually defines an operation between two DPs which will be discussed in more detail in the following. Note that extending the above two-layer HDP to more layers is straightforward under this definition. It was originally designed to model *group data*. For example, there are D documents (i.e., groups) and each G_d could be adopted to model one document using the mixture idea in (Rasmussen, 1999), such as

$$x_i \sim F(\theta_i), \quad \theta_i \sim G \tag{6.3}$$

where x_i is a data point generated according to a distribution parameterized by a draw θ_i from G.

To capture the cooperative hierarchical structures, two operations between random measures from DP are formally defined as follows

Definition 11 (Inheritance) *A probability measure G_1 is the* Inheritance *from another probability measure G_2 from DP on space Θ by taking G_2 as its base measure*

$$G_1 \sim DP(\alpha_1, G_2), \quad G_2 \sim DP(\alpha_2, H) \tag{6.4}$$

where α_1 and α_2 are DP parameters. The discrete nature of G_2 enables G_1 to *inherit factors/atoms from G_2.*

Note that this operation is from HDP.

Definition 12 (Cooperation) *A measure G is the* Cooperation *of two probability measures, i.e., G_1 and G_2, from DP on the same space Θ, if*

$$G = G_1 \oplus G_2 \tag{6.5}$$

where G is a new probability measure on the space Θ. For any given partition $\{\Omega\}_{k=1}^{\infty}$ on Θ, it has

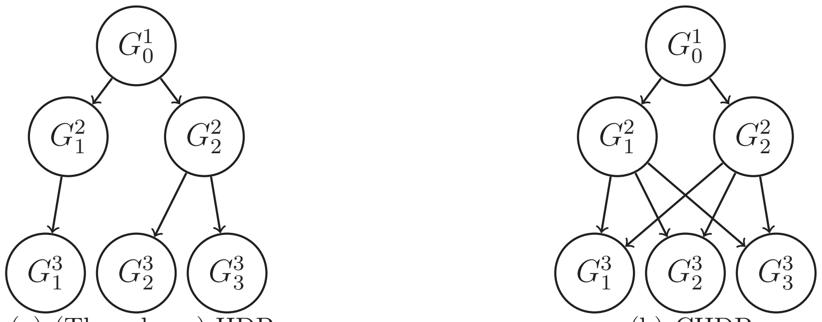
$$G(\Omega_k) = G_1(\Omega_k) \oplus G_2(\Omega_k) \tag{6.6}$$

where $\oplus$ denotes the convex combination. Extending the Cooperation *of more than two probability measures is straightforward.*

Theorem 1 *The Cooperation of finite number of probability measures is still a probability measure.*

Proof 2 *A probability measure $\mathbb{P}$ is a real-valued function, defined on the collection of events, which should satisfy the following axioms: 1) $\mathbb{P}(\Omega_k) \geq 0$ for every event Ω_k; 2) $\mathbb{P}(\Theta) = 1$; 3) If $\{\Omega_k\}$ is countable and disjoint, then $\mathbb{P}(\bigcup_k \Omega_k) = \sum_k \mathbb{P}(\Omega_k)$.*

Suppose there are a finite number of probability measures $\{G_i\}_{i=1}^{I}$ and G is their Cooperation *with combination weights $\{\varpi_i\}_{i=1}^{I}$ and $\sum_{i=1}^{I} \varpi_i = 1$. For every event Ω_k, each G_i satisfies $G_i(\Omega_k) \geq 0$, so $G(\Omega_k) = \sum_i \varpi_i G_i(\Omega_k) \geq 0$; For the entire sample space Θ, each G_i satisfies $G_i(\Theta) = 1$, so $G(\Theta) = \sum_i \varpi_i G_i(\Theta) = 1$; Since each measure G_i in Eq. (6.5) is with the same countable set, an injective function exists from this countable set to the natural numbers. Therefore, the element pairwise superposition in Eq. (6.6) is well defined. Similarly, $G(\bigcup_k \Omega_k) = \sum_i \varpi_i G_i(\bigcup_k \Omega_k) = \sum_i \varpi_i \sum_k G_i(\Omega_k) = \sum_k \sum_i \varpi_i G_i(\Omega_k) = \sum_k G(\Omega_k)$.*



(a) (Three-layer) HDP (b) CHDP

Figure 6.2: The comparison between generated random measures from HDP and CHDP.

With both *Inheritance* and *Cooperation* in hand, the cooperative hierarchical Dirichlet process is defined as follow

Definition 13 (Cooperative Hierarchical Dirichlet Process) *A* coopera-tive hierarchical Dirichlet process (CHDP) *(Teh, 2006) is a distribution over a set of random probability measures (over Θ) located at multiple layers as shown in Fig. 6.2(b). The process defines:*

- *Each layer is with a number N^ℓ of random probability measures $\{G_i^\ell\}_{i=1:N^\ell}$ where $N^1 = 1$ for the first layer;*

- *At the first layer $\ell = 1$, a single global random probability measure G_0 is defined, which is distributed as a Dirichlet process parameterized by a concentration parameter α and a base probability measure H*

$$G_0 \sim DP(\alpha, H) \tag{6.7}$$

- *At the following layer $\ell > 1$, each probability measure G_i^ℓ at layer ℓ is the* Inheritance *from the* Cooperation *of all probability measures at the upper layer $\ell - 1$*

$$G_i^\ell \sim DP(\alpha_\ell, G^{\ell-1}) \tag{6.8}$$

where

$$G^{\ell-1} = G_1^{\ell-1} \oplus G_2^{\ell-1} \oplus \cdots \oplus G_{N^{\ell-1}}^{\ell-1} \tag{6.9}$$

and α_ℓ *is the DP parameter at the layer* ℓ*.*

In the Fig. 6.2(a) and 6.2(b), the generated sets of random measures from both HDP and CHDP are compared. The similarity is that the random measures in the second layer are used as the base measures of the DP to generate measures in the third layer. The difference is the random measures at the second and third layers of the HDP, shown in Fig. 6.2(a), have a one-to-many relationship, where Fig. 6.2(b) (or CHDP) shows a many-to-many relationship.

For clarity, the node at the first layer means the G_0^1 of CHDP prior in Fig. 6.2(b) but not the (red) nodes of CHS in Fig. 6.1(b); a node at the second layer means a G^2 of CHDP prior in Fig. 6.2(b) which corresponds to a (red) node of CHS in Fig. 6.1(b); a node at the third layer means the G^3 of CHDP prior in Fig. 6.2(b) which corresponds to an (orange) node of CHS in Fig. 6.1(b) in the rest of paper.

It can be seen from the Definition 13 and Fig. 6.2 that HDP can be seen as a special case of CHDP with each child node/probability measure having only one parent node/probability measure. If the cooperative/Type-II hierarchical structure degenerates into Type-I hierarchical structure , the CHDP will degenerate into a HDP as well.

This study aims to model cooperative hierarchical structures using CHDP-based model. Based on the Fig. 6.2(b), the following discussion explains how CHDP is used to model an *author-paper-word* structure. Suppose the papers are from the area of *Artificial Intelligence*. G_0^1 at the top layer defines a set of topics within these papers which may be *machine learning*, *constraint processing*, *intelligent robotics*, and so on. There are two random measures G_1^2 and G_2^2 at the second layer for two authors, and G_1^2 denotes the interest of first author on above topics. Two authors have cooperatively written three papers. G^3_1, G^3_2 and G^3_3, at the third layer, represent the ratios of topics in three papers respectively. Combining the mixture in Eq. (6.3), the words in each paper could be generated where $F()$ is set as a multinomial distribution with a topic as the parameter. Above procedure could also be seen as the generative process of the CHDP-based topic model. With the observed data, it could inference all the random measures in the Fig. 6.2(b). In the remainder of this paper, the above *author-paper-word* structure and Fig. 6.2(b) are kept as the example to explain the inference of CHDP, so the following do not distinguish: node a with author a and node d with document d.

Next, two constructive representations for CHDP are introduced: stick-breaking representation (explicit one) and international restaurant process representation (marginal one), and corresponding inference algorithms for CHDP based on them.

6.3.1 Stick-breaking representation

Analogous to the stick-breaking representations of DP and HDP (Sethuraman, 1994), a stick-breaking representation for the proposed CHDP is introduced and an inference algorithm for CHDP is developed based on this representation.

Description

Based on the stick-breaking process for HDP (Sethuraman, 1994; Wang et al., 2011), the following stick-breaking representation for CHDP has been developed

$$G_0 = \sum_{k}^{\infty} \pi_{0,k} \delta_{\theta_k} \qquad \pi_{0,k} = \nu_{0,k} \prod_{j=1}^{k-1} (1 - \nu_{0,j}) \qquad \nu_{0,k} \sim Beta(1, \alpha)$$

$$\begin{array}{lll} G_a = \sum_o^{\infty} \pi_{a,o}\delta_{\theta_{a,o}} & \pi_{a,o} = \nu_{a,o}\prod_{j=1}^{o-1}(1-\nu_{a,j}) & \nu_{a,o} \sim Beta(1,\alpha_a) \\ G_d = \sum_t^{\infty} \pi_{d,t}\delta_{\theta_{d,t}} & \pi_{d,k} = \nu_{d,t}\prod_{j=1}^{t-1}(1-\nu_{d,j}) & \nu_{d,t} \sim Beta(1,\alpha_d) \\ \theta_k \sim H & \theta_{a,o} = \theta_{z_{a,o}} & z_{a,o} \sim \pi_0 \\ \theta_{d,t} = \theta_{z_{d,t}} & z_{d,t} \sim \left\{\varpi_{d,a_1}\pi_{a_1}, \cdots, \varpi_{d,a_A}\pi_{a_A}\right\} & \end{array}$$

where A is the number of nodes at the second layer and $\varpi_{d,a}$ is the weight of link between node a and node d.

Note that the *Inheritance* and *Cooperation* take place between the stick weights π of the nodes. The difficult part of this model is to infer π_a considering their *Cooperation*. Since it needs to sample $z_{d,t}$ from $\{\varpi_{d,a_1}\pi_{a_1}, \cdots, \varpi_{d,a_A}\pi_{a_A}\}$, it is worth to note the following property of this set

Theorem 2 *The set $\{\varpi_{d,a_1}\pi_{a_1}, \cdots, \varpi_{d,a_A}\pi_{a_A}\}$ is a countable infinite set and the summation of all elements is equal to one.*

Proof 3 *This set is the union of the stick weights (with an adjustment) from all the nodes in the second layer linked with the node d. Since stick weights of a node in the second layer are a countable set, the union of finite number of countable sets is still a countable set.*

Since the stick weights $\{\pi_{a,o}\}$ of each node at the second layer is from a DP, their summation is definitely equal to one $\sum_{o=1}^{\infty}\pi_{a,o} = 1$. Considering the adjustment $\varpi_{d,a}$, the summation becomes $\sum_{o=1}^{\infty}\varpi_{d,a}\pi_{a,o} = \varpi_{d,a}$. Therefore, the summation of finite number A of these stick weights is equal to 1 because $\sum_a \varpi_{d,a} = 1$.

Note that there is no one-to-one mapping between $\pi_{a,o}$ with $\pi_{a,k}$. In fact, their relation is $\pi_{a,k} = \sum_{o:z_{a,o}=k}\pi_{a,o}$. Similar with $\pi_{d,k}$ and $\pi_{d,t}$, their relation is

$\pi_{d,k} = \sum_{t:z_{z_{d,t}}=k} \pi_{d,t}$. Next, the inference method based on this representation is given.

Inference

This section has designed a Markov Chain Monte Carlo (Andrieu et al., 2003) algorithm to obtain samples of posterior distribution given the observation $p(\pi_0, \pi_a, \pi_b, \theta | data)$, however combining *Inheritance* and *Cooperation* presents a challenge, and the infinite number of factors and their weights make the posterior inference of the latent variables even harder. One common work-around in nonparametric Bayesian learning is to use a truncation method. The truncation method (Fox, 2009; Willsky et al., 2009), which uses a relatively big $K^\dagger$ as the (potential) maximum number of topics, is widely accepted but brings an approximation error. Another successful technique to resolve this problem is: *data/variable augmentation* (Van Dyk and Meng, 2001; Tanner and Wong, 2010), also known as *Slice Sampling* (Damlen et al., 1999; Neal, 2003*b*). Therefore, an auxiliary variable-based inference algorithm has been developed to make inference feasible. Next, the sampling of each variable is given in detail.

Sampling $\mathbf{u_{d,n}}$. $u_{d,n}$ is an introduced auxiliary variable that can make the possible assigned factors to data n of node d within a finite scope, which can also be known as *adaptive truncation*

$$u_{d,n} \sim Unif(0, \pi_{d,z_{d,n}}) \tag{6.10}$$

where $z_{d,n}$ is the former assignment of n and $Unif(0, \pi_{d,z_{d,n}})$ denotes a uniform distribution with support on $[0, \pi_{d,z_{d,n}}]$.

Sampling $\pi_{\mathbf{d,t}}$. The update of the stick weights of node d should consider the adaptive truncation $u_{d,n}$ (i.e., the inequalities between slice variables and the stick weights). In (Neal, 2003*b*; Kalli et al., 2011), it has been proofed that the stick breaks satisfy truncated Beta distributions. For $t \leq t^*$,

$$\nu_{d,t} \sim beta(\nu_{d,t}; 1, \alpha_d) \cdot \mathbb{1}(L_t < \nu_{d,t} < U_t) \tag{6.11}$$

where

$$\begin{aligned} t^* &= \max_n \{z_{d,n}\} \\ L_t &= \max_{n: z_{d,n}=t} \left\{ \frac{u_{d,n}}{\prod_{l:l<t}(1-\nu_{d,l})} \right\} \\ U_t &= 1 - \max_{n: z_{d,n}>t} \left\{ \frac{u_{d,n}}{\nu_{d,z_{d,n}} \prod_{l:l<z_{d,n}, l \neq t}(1-\nu_{d,l})} \right\} \end{aligned} \tag{6.12}$$

where $\mathbb{1}()$ is an indicator function parameterized by a condition and equals 1 if the condition is satisfied; 0, otherwise, and t^* is the maximum number of used factors of node/document d by the words in this document. Note that it is possible to only update these breaks, a finite number, by leaving the others, an infinite number, not updated, because the posteriors of other weights are same as their priors. Since there is a determinant relation between stick breaks and stick weights, the update of breaks will update weights at the same time.

Sampling $\mathbf{z_{d,n}}$. $z_{d,n}$ is an indicator variable to express the assigned factor to word n of document d. The posterior conditional distribution is

$$p(z_{d,n} = t | \cdots) \propto \mathbb{1}(u_{d,n} < \pi_{d,t}) \cdot \theta_{z_{a,z_{d,t}}, v_{d,n}} \tag{6.13}$$

where $v_{d,n}$ is the word name of n-th word in document d and $\theta_{z_{z_{d,t}}, v_{d,n}}$ is the likelihood of assignment. The *adaptive truncation* is reflected by the sampling of the $z_{d,n}$. Only the factors with larger weights than $u_{d,n}$ are considered as the assignment candidates. Since the value of $u_{d,n}$ is larger than zero, this number must be finite.

Sampling $\mathbf{u_{d,t}}$. $u_{d,t}$ also functions as an adaptive truncation, but it is a little more complicated than $u_{d,n}$ because of the *Cooperation* between the measures in the second layer. Therefore the following slice variable is introduced

$$u_{d,t} \sim Unif(0, \varpi_{d,a}\pi_{z_{d,t}}) \tag{6.14}$$

where $\varpi_{d,a}$ is the weight of link between node a and node d.

Sampling $\pi_{\mathbf{a,o}}$. Based on the adaption truncation $u_{d,t}$, update the stick breaks of the nodes at the second layer as follows: for $o \leq o^*$,

$$\nu_{a,o} \sim beta(\nu_{a,o}; 1, \alpha_a) \cdot \mathbb{1}(L_o < \nu_{a,o} < U_o) \tag{6.15}$$

where

$$\begin{aligned} o^* &= \max_{(d,t)}\{z_{d,t}\} \\ L_o &= \max_{(d,t):z_{d,t}=(a,o)} \left\{ \frac{u_{d,t}}{\varpi_{d,a}\prod_{l:l<o}(1-\nu_{a,l})} \right\} \\ U_o &= 1 - \max_{(d,t):z_{d,t}>o} \left\{ \frac{u_{d,t}}{\varpi_{d,a}\nu_{a,z_{d,t}}\prod_{l:l<o,l\neq o}(1-\nu_{a,l})} \right\} \end{aligned}$$

where L_o needs to consider all nodes at the third layer linked with node a.

Sampling $\mathbf{z_{d,t}}$. $z_{d,t}$ is an indicator variable to express the assigned factor of a node at the second layer to the t-th factor of node d. The posterior conditional distribution is

$$p(z_{d,t} = (a, o)|\cdots) \propto \mathbb{1}(u_{d,t} < \varpi_{d,a}\pi_{a,o}) \cdot \prod_{n:z_{d,n}=t} \theta_{z_{z_{d,t}}, v_{d,n}} \tag{6.16}$$

where $z_{d,t}$ is an index that means the t-th factor of node d is assigned to o-th factor of node a.

Sampling $\mathbf{u_{a,o}}$. $u_{a,o}$ is the slice variable for the random measures at the second layer which is similar to $u_{d,n}$, and satisfies

$$u_{a,o} \sim U(0, \pi_{0,z_{a,o}}) \tag{6.17}$$

Sampling $\pi_{\mathbf{0},\mathbf{k}}$. The update of the stick breaks of the global measure G_0. For $k \leq k^*$,

$$\nu_{0,k} \sim beta(\nu_{0,k}; 1, \alpha_0) \cdot \mathbb{1}(L_k < \nu_{0,k} < U_k) \tag{6.18}$$

where

$$\begin{aligned} k^* &= \max_{(a,o)}\{z_{a,o}\} \\ L_k &= \max_{(a,o):z_{a,o}=k} \left\{ \frac{u_{a,o}}{\prod_{l:l<z_{a,o}}(1-\nu_{0,l})} \right\} \\ U_k &= 1 - \max_{(a,o):z_{a,o}>k} \left\{ \frac{u_{a,o}}{\nu_{0,z_{a,o}} \prod_{l:l<z_{b,t},l\neq k}(1-\nu_{0,l})} \right\} \end{aligned}$$

Sampling $\mathbf{z}_{\mathbf{a},\mathbf{o}}$. This indicator variable is to indicate the assigned node to o-th factor at the second layer. The posterior conditional distribution is

$$p(z_{a,o} = k | \cdots) \propto \mathbb{1}(u_{a,o} < \pi_{0,k}) \cdot \prod_{(d,n):z_{d,n}=t, z_{d,t}=(a,o)} \theta_{k,v_{d,n}} \tag{6.19}$$

Sampling $\theta_{\mathbf{k}}$. θ_k denotes a global factor and its posterior distribution is

$$p(\theta_k | \cdots) \propto Dir(\theta_k; \gamma) \cdot \prod_{(d,n):z_{d,n}=t, z_{d,t}=o, z_{a,o}=k} \theta_{k,v_{d,n}} \tag{6.20}$$

where $Dir(\theta_k; \gamma)$ is the prior H and γ is the hyper-parameter.

Next, the reasonability of introducing the auxiliary variables is shown using $u_{d,t}$ as an example because it is the most complicated and the others follow if it is verified.

Theorem 3 *Introducing the slice variable $u_{d,t}$ with the distribution in Eq. (6.14)*

Algorithm 6.1: Slice Sampler for CHDP

initialization;
do
  for $d = 1; d \leq D$ **do**
    for $n = 1; n \leq N_d$ **do**
      Update $u_{d,n}$ by Eq. (6.10);
    Update $\pi_{d,t}$ by Eq. (6.11);
    for $n = 1; n \leq N_d$ **do**
      Update $z_{d,n}$ by Eq. (6.13);
    for $t = 1; t \leq T_d$ **do**
      Update $u_{d,t}$ by Eq. (6.14);
  for $a = 1; a \leq A$ **do**
    for $o = 1; a \leq O_a$ **do**
      Update $\pi_{a,o}$ by Eq. (6.15);
      Update $u_{a,o}$ by Eq. (6.17);
  for $d = 1; d \leq D$ **do**
    for $t = 1; t \leq T_d$ **do**
      Update $z_{d,t}$ by Eq. (6.16);
  for $k = 1; k \leq K$ **do**
    Update $\pi_{0,k}$ by Eq. (6.18);
  for $a = 1; a \leq A$ **do**
    for $o = 1; a \leq O_a$ **do**
      Update $z_{a,o}$ by Eq. (6.19);
  for $k = 1; k \leq K$ **do**
    Update θ_k by Eq. (6.20);
while *convergent*;
return K, $\{\theta_k\}_{k=1}^K$, $\{\pi_a\}_{a=1}^A$, $\{\pi_d\}_{d=1}^D$;

will not change the posterior distribution of $z_{d,t}$

$$p(z_{d,t} = (a, o)|\cdots) = \varpi_{d,a}\pi_{a,o} \cdot \prod_{n:z_{d,n}=t} \theta_{z_{z_{d,t}},v_{d,n}} \tag{6.21}$$

Proof 4 *Here, check the marginal distribution of* $z_{d,t}$ *after integrating out the* *newly introduced slice variables*

$$
\begin{aligned}
p(z_{d,t}|\cdots) &= \int_{u_{d,t}} p(z_{d,t}, u_{d,t}|\cdots)\mathrm{d}u_{d,t} \\
&= \int_{u_{d,t}} p(z_{d,t}|u_{d,t}, \cdots) \cdot p(u_{d,t}|\cdots)\mathrm{d}u_{d,t} \\
&= \varpi_{d,a}\pi_{a,o} \cdot \prod_{(d,n):z_{d,n}=t} \theta_{z_{z_{d,t}},v_{d,n}}
\end{aligned}
\tag{6.22}
$$

It is exactly same as Eq. (6.21).

The entire update procedure is summarized in Algorithm 6.1. The physical meanings of these variables are explained in the next subsection with a metaphor.

6.3.2 International restaurant process representation

The international restaurant process (IRP) representation for CHDP is described, then the designed inference method based on this representation is introduced.

Description

The marginal representation of CHDP with G_0, $\{G_a\}_{a=1}^{A}$, and $\{G_d\}_{d=1}^{D}$ marginalized out (named international restaurant process) is as follows

$$
\begin{aligned}
G_a^d &= \varpi_{d,a_1}G_{a_1} + \varpi_{d,a_2}G_{a_2} + \cdots + \varpi_{d,A}G_A \\
\theta_{d,n}|\theta_{d,1}, \cdots, \theta_{d,n-1}, G_a^d &\sim \sum_{t=1}^{T_d} \frac{N_{d,t}}{\sum_t N_{d,t} + \alpha_d}\delta_{\theta_{d,t}} + \frac{\alpha_d}{\sum_t N_{d,t} + \alpha_d} G_a^d
\end{aligned}
\tag{6.23}
$$

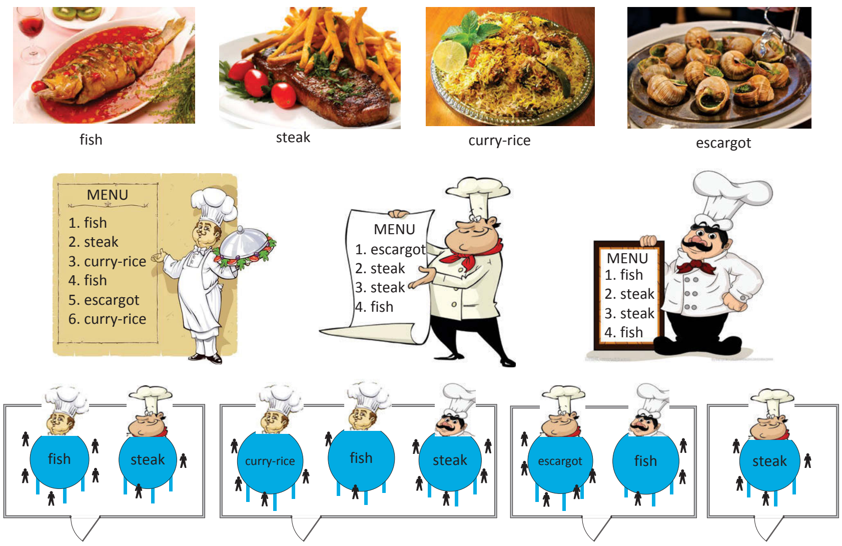
$$
\theta_{d,t}|\theta_{1,1}, \cdots, \theta_{d,t-1}, G_0 \sim \sum_{o=1}^{O_a} \frac{T_{a,o}}{\sum_o T_{a,o} + \alpha_a}\delta_{\theta_{a,o}} + \frac{\alpha_a}{\sum_o T_{a,o} + \alpha_a} G_0 \tag{6.24}
$$

$$\theta_{a,o}|\theta_{1,1},\cdots,\theta_{a,o-1},H \sim \sum_{k=1}^{K} \frac{O_k}{\sum_k O_k + \alpha_0}\delta_{\theta_k} + \frac{\alpha_0}{\sum_k O_k + \alpha_0}H \tag{6.25}$$

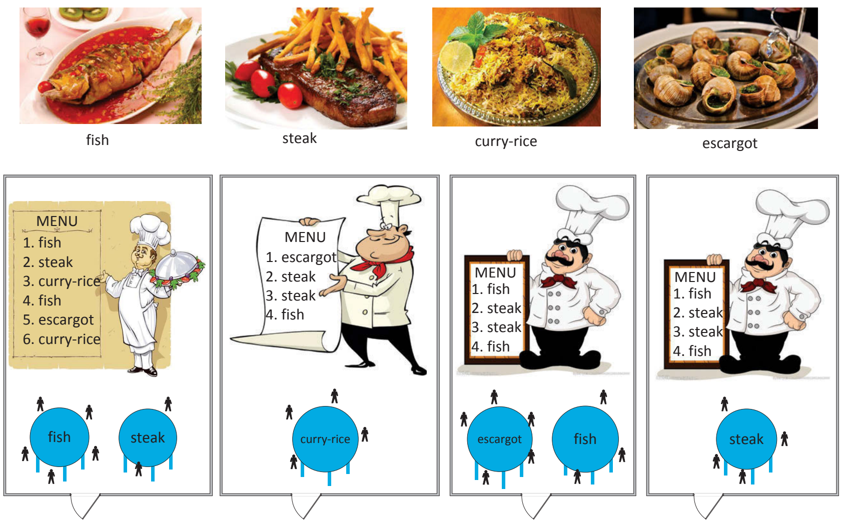
where $N_{d,t}$ denotes the number of $\theta_{d,n}$ associated with $\theta_{d,t}$ in d; $T_{a,o}$ denotes the number of $\theta_{d,t}$ associated with $\theta_{a,o}$; and O_k denotes the number of $\theta_{a,o}$ associated with θ_k.

Similar to the Chinese restaurant process of DP outlined Section 2.1.2 and the Chinese restaurant franchise in HDP in Section 2.2.2, a metaphor eases understanding of IRP. Since CHDP is based on a three-layer HDP, the following paragraphs describe the metaphor for three-layer HDP first, and then introduces the one for CHDP. Note that the CRF in Section 2.2.2 is only a two-layer HDP.

As shown in Fig. 6.3(b), the metaphor for the three-layer HDP is as follows: there is a global menu with different dishes $\{\theta_k\}_{k=1}^K$ shared by all the chefs $\{a\}$ from different countries (i.e., China, India, Italy, France). Each chef has a personal menu with dish names as menu options $\{\theta_{a,o}\}$ (Note that menu options are not eliminative - different options could, in fact, be the same dish.) according to her preference and ability. There are also several (national) restaurants $\{d\}$. Each restaurant has employed one (and only one) chef, but a chef could work in different restaurants. For example, a French restaurant hired a French chef, but this chef may work in other French restaurants. In each restaurant, there are multiple tables T_d, and each table is served with a dish cooked by the chef of this restaurant. When a customer n walks into a restaurant d, she sits at an occupied table with the probability $\frac{N_{d,t}}{\sum_t N_{d,t}+\alpha_d}$ or a new table with the probability $\frac{\alpha_d}{\sum_t N_{d,t}+\alpha_d}$. If an occupied table is selected, she just eats the dish on this table; if the table is new, the customer needs to order a dish for this table from the personal menu of the chef. If option o on the menu is selected with the probability $\frac{T_{a,o}}{\sum_o T_{a,o}+\alpha_a}$, she eats it; if she is not satisfied by all the current options on the menu with the probability $\frac{\alpha_a}{\sum_o T_{a,o}+\alpha_a}$, the chef has to add a new option on the menu



(a) International Restaurant Process (IRP)



(b) (Three-layer) Chinese Restaurant Franchise (CRF)

Figure 6.3: Comparing the Chinese restaurant franchise process (three-layer) and international restaurant process. There are four restaurants and three chefs in the figure. In CRF, all the customers in a restaurant can only be served by one chef, but the customers in IRP could be served by different chefs. The main difference between HDP and CHDP is due to *Cooperation*.

from the global shared menu. If dish k on the global menu is selected with the probability $\frac{O_k}{\sum_k O_k+\alpha_0}$, she eats it; if all the dishes on the global menu still cannot satisfy this customer with the probability $\frac{\alpha_0}{\sum_k O_k+\alpha_0}$, the chefs have to add a new dish to this global menu (while embarrassedly looking up a recipe book H).

As shown in Fig. 6.3, the metaphor for the CHDP is: there is a global menu with different dishes $\{\theta_k\}_{k=1}^{K}$ shared by all the chefs $\{a\}$ from different countries (i.e., China, India, Italy, France). Each chef has a personal menu with dish names as menu options $\{\theta_{a,o}\}$ (again, different options could be same dish). There are also several (international) restaurants $\{d\}$ served by these chefs. Each restaurant could employ a number of chefs from different countries, and a chef could work in different restaurants as well. For example, an international restaurant may have a Chinese chef, a French chef, an Italian chef, and an Indian chef (hence its name, *international restaurant*). In each restaurant, there are multiple tables T_d with a dish cooked by the chefs employed by this restaurant. When a customer n walks into an international restaurant d, she sits at an occupied table with the probability $\frac{N_{d,t}}{\sum_t N_{d,t}+\alpha_d}$ or a new table with the probability $\frac{\alpha_d}{\sum_t N_{d,t}+\alpha_d}$. If an occupied table is selected, she just eats the dish on this table; if the table is new, the customer needs to order a dish for this table from the menus of all the chefs working in this restaurant. If option o on the menu of a chef a is selected with the probability $\frac{T_{a,o}}{\sum_o T_{a,o}+\alpha_a}$, she eats it; if she is not satisfied by the current options on the menu with the probability $\frac{\alpha_a}{\sum_o T_{a,o}+\alpha_a}$, she can ask this chef a to add a new option on his menu from the global shared menu. If dish k on the global menu is added to his menu with the probability $\frac{O_k}{\sum_k O_k+\alpha_0}$, she eats it; if all the dishes on the global menu cannot satisfy this customer with the probability $\frac{\alpha_0}{\sum_k O_k+\alpha_0}$, the chefs have to add a new dish to this global menu.

In this metaphor, the connection to the cooperative hierarchical structures (e.g., *author-paper-word*) and CHDP is as follows: a dish θ_k corresponds to a topic (a factor of global measure G_0 at the first layer in Fig. 6.2(b)); a chef a corresponds to an *author* (a node a in the second layer in Fig. 6.2(b)) and an menu option of chef a corresponds to a factor $\theta_{a,o}$ of the measure G^a for node a; a restaurant d corresponds to a *document* (a node d in the third layer in Fig. 6.2(b)) and a table of restaurant d corresponds to a factor $\theta_{d,t}$ of the measure G^d for node d; a customer n corresponds to a *word*.

Inference

Next, a Markov Chain Monte Carlo algorithm is designed to obtain samples of posterior distribution $p(\{\theta_k\}, \{\theta_{a,o}\}, \{\theta_{d,t}\}, K|data, \cdots)$ of CHDP based on IRP representation. The final aim is to learn the factors and the representations of all the nodes on these factors, although combining *Inheritance* and *Cooperation* still presents difficulties.

Sampling $\theta_{d,n}$. Its prior is as in Eq. (6.23). Since G_a^b is different from the one in HDP and hard to marginalize out, an auxiliary variable $\hat{z}_{d,n}$ is introduced to make it inferrable. $\hat{z}_{d,n}$ denotes the selected chef of customer n in restaurant d. If it is known which chef this customer selects, it is just to assign a dish to him by marginalizing the probability measure of the selected chef. However, it must make sure that the introduced auxiliary variable will not change the whole posterior distribution. Here, the distribution of the auxiliary variable $\hat{z}_{d,n}$ is defined as

$$p(\hat{z}_{d,n} = a|\cdots) = \frac{N_{d,t}^a + \varpi_{d,a}\alpha_d}{\sum_t N_{d,t} + \alpha_d} \tag{6.26}$$

where $N_{d,t}^a$ denotes the number of customers on table t served by chef a in restaurant d. With the selected chef, $\theta_{d,n}$ can be sampled by

$$\theta_{d,n}|\hat{z}_{d,n} = a, G_a, \cdots \sim \sum_{t=1}^{T_d^a} \frac{N_{d,t}}{N_{d,t}^a + \varpi_{d,a}\alpha_d}\delta_{\theta_{d,t}} + \frac{\varpi_{d,a}\alpha_d}{N_{d,t}^a + \varpi_{d,a}\alpha_d} G_a \tag{6.27}$$

where T_d^a is the table number in restaurant d served by chef a and $t \in a$ denotes table t served by chef a. If a new dish is needed, it needs to sample from G_a.

Theorem 4 *Introducing the auxiliary variable $\hat{z}_{d,n}$ with the distribution in Eq. (6.26) will not change the final posterior distribution.*

Proof 5 *The marginal distribution of $\theta_{d,n}$ with $\hat{z}_{d,n}$ marginalized out is*

$$
\begin{aligned}
p(\theta_{d,n}) &= \sum_{\hat{z}_{d,n}} p(\theta_{d,n}, \hat{z}_{d,n}) p(\hat{z}_{d,n}) \\
&= \sum_{a} p(\theta_{d,n} | \hat{z}_{d,n} = a) p(\hat{z}_{d,n} = a) \\
&= \sum_{a} \left(\sum_{t=1}^{T_d^a} \frac{N_{d,t}}{N_{d,t}^a + \varpi_{d,a}\alpha_d} \delta_{\theta_{d,t}} + \frac{\varpi_{d,a}\alpha_d}{N_{d,t}^a + \varpi_{d,a}\alpha_d} G_a \right) \cdot \frac{N_{d,t}^a + \varpi_{d,a}\alpha_b}{\sum_t N_{d,t} + \alpha_d} \\
&= \sum_{a} \left(\sum_{t=1}^{T_b^a} \frac{N_{d,t}}{\sum_t N_{d,t} + \alpha_d} \delta_{\theta_{d,t}} + \frac{\varpi_{d,a}\alpha_d}{\sum_t N_{d,t} + \alpha_d} G_a \right) \\
&= \sum_{t=1}^{T_d} \frac{N_{d,t}}{\sum_t N_{d,t} + \alpha_d} \delta_{\theta_{d,t}} + \frac{\alpha_d}{\sum_t N_{d,t} + \alpha_d} \left(\varpi_{d,a_1} G_{a_1} + \varpi_{d,a_2} G_{a_2} + \cdots \right) \\
&= \sum_{t=1}^{T_d} \frac{N_{d,t}}{\sum_t N_{d,t} + \alpha_d} \delta_{\theta_{d,t}} + \frac{\alpha_d}{\sum_t N_{d,t} + \alpha_d} G_a^d
\end{aligned}
$$

The result is same as in Eq. (6.23). So one conclusion is drawn that introducing an auxiliary variable will not impact on the posterior distribution of the $\theta_{d,n}$.

In the MCMC algorithm, the posterior distribution is approximated by its samples, which is the product of the prior and likelihood. The above demonstrates the prior, and the likelihood is derived by

$$L(\theta_{d,n}) \propto \begin{cases} F(v_{d,n}|\theta_{d,t}) & \text{if t is occupied} \\ \sum_{o=1}^{O_a} \frac{T_{a,o}}{\sum_o T_{a,o}+\alpha_a} F(v_{d,n}|\theta_{a,o}) + \frac{\alpha_a}{\sum_o T_{a,o}+\alpha_a} LC(v_{d,n}) & \text{if t is new} \end{cases} \tag{6.28}$$

where $v_{d,n}$ denotes the customer n in restaurant d and

$$\begin{aligned} LC(v_{d,n}) &= \sum_{k=1}^{K} \frac{O_k}{\sum_k O_k + \alpha_0} F(v_{d,n}|\theta_k) + \frac{\alpha_0}{\sum_k O_k + \alpha_0} LK(v_{d,n}) \\ LK(v_{d,n}) &= \int_\theta F(v_{d,n}|\theta) H(\theta) \mathrm{d}\theta \end{aligned}$$

and O_k is the number options with dish k; F is a multinomial distribution with θ as a parameter; and H is a V-dimensional Dirichlet distribution parameterized by γ. Due to the conjugate relationship between the multinomial and Dirichlet distributions, there is a closed-form for LK.

The remaining posterior distributions simply follow the three-layer HDP. For the self-contained reason, the likelihood parts of remaining variables are also listed below.

Sampling $\theta^a_{d,t}$. For assigning an menu option o to each table served by chef a in restaurant d, the prior for $\theta^a_{d,t}$ is as the one in Eq. (6.24) and the likelihood part is

$$L(\theta^a_{d,t}) \propto \begin{cases} F(v_{d,t}|\theta_{a,o}) & \text{if } o \text{ is occupied} \\ \sum_{k=1}^{K} \frac{O_k}{\sum_k O_k+\alpha_0} F(v_{d,t}|\theta_k) + \frac{\alpha_0}{\sum_k O_k+\alpha_0} LK(v_{d,t}) & \text{if } o \text{ is new} \end{cases} \tag{6.29}$$

where $v_{d,t}$ denotes all the customers sitting on the table t in restaurant d, O_k is the number menu options with dish k

$$LK(v_{d,t}) = \int_{\theta} F(v_{d,t}|\theta)H(\theta)\mathrm{d}\theta.$$

Sampling $\theta_{a,o}$ For assigning a dish k to each menu option o of chef a, the prior is as the one in Eq. (6.25) and the likelihood part is

$$L(\eta_{a,o}) \propto \begin{cases} F(v_{a,o}|\theta_k) & \text{if } k \text{ is occupied} \\ LK(v_{a,o}) & \text{if } k \text{ is new} \end{cases} \tag{6.30}$$

where $v_{a,o}$ denotes all the customers served by the o-th menu option of chef a

$$LK(x_{a,o}) = \int_{\theta} F(v_{a,o}|\theta)H(\theta)\mathrm{d}\theta.$$

Sampling $\theta_{\mathbf{k}}$. θ_k denotes a global factor/topic and its posterior distribution is

$$p(\theta_k|\cdots) \propto Dir(\theta_k;\gamma) \cdot F(v_k|\theta_k) \tag{6.31}$$

where v_k is total number of customers assigned to k.

The whole procedure for the inference of IRP is summarized in Algorithm 6.2.

6.3.3 Model analysis

The Bayesian nonparametric models (i.e., different stochastic processes or their designed combinations) actually provide a prior for the number of factors. Apparently, the expected factor number from this prior is determined by the parameters of the designed nonparametric priors, so it is necessary to investigate the relationships between the model parameters with the expectation of the factor

Algorithm 6.2: Gibbs Sampler for IRP

```
initialization;
do
    for d = 1; d ≤ D do
        for n = 1; n ≤ N_d do
            Update ẑ_{d,n} by Eq. (6.26);
            Update θ_{d,n} by Eq. (6.23) and Eq. (6.28);
        for t = 1; t ≤ T_d do
            Update θ^a_{d,t} by Eq. (6.24) and Eq. (6.29);
    for a = 1; a ≤ A do
        for o = 1; o ≤ O_a do
            Update θ_{a,o} by Eq. (6.25) and Eq. (6.30);
    Update θ by Eq. (6.31);
while convergent;
return K, {θ_k}_{k=1}^K, {{θ_{d,t}}_{t=1}^{T_d}}_{d=1}^D}, {{θ_{a,o}}_{o=1}^{O_a}}_{a=1}^A ;
```

number. For the proposed CHDP, the following result holds

Theorem 5 *Given a (three-layer) cooperative hierarchical structure, the expected factor number from the CHDP is*

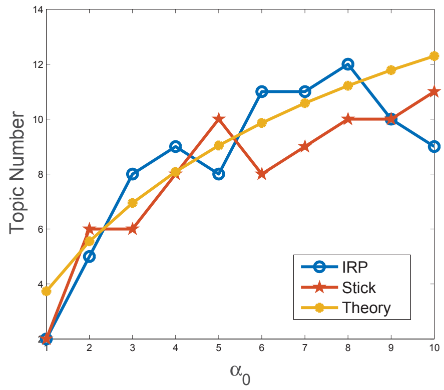
$$\mathbb{E}[K] = \sum_{o=1}^{\sum_a O_a} \frac{\alpha_0}{o - 1 + \alpha_0} \tag{6.32}$$

where

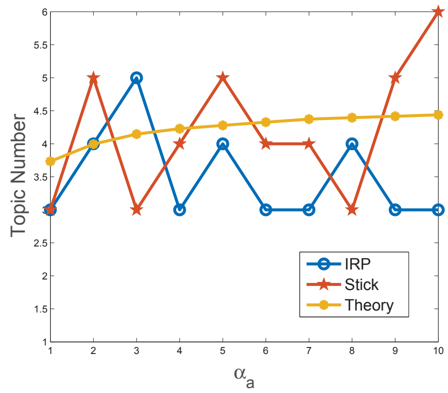
$$O_a = \sum_{t=1}^{T_a} \frac{\alpha_a}{t - 1 + \alpha_a}, \quad T_a = \sum_{d}^{D} \varpi_{d,a} T_d, \quad T_d = \sum_{n=1}^{N_d} \frac{\alpha_d}{n - 1 + \alpha_d} \tag{6.33}$$

and $\varpi_{d,a}$ is the weight of link between node a at the second layer and node d at the third layer; α_0, α_a and α_d are three parameters of the CHDP.

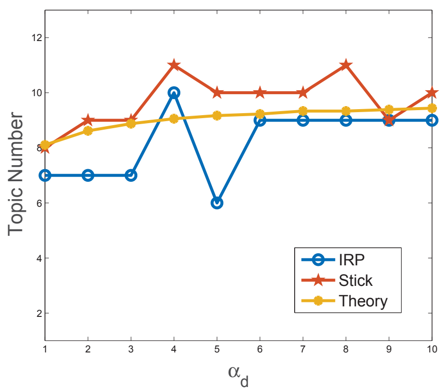
Proof 6 *CHDP is composed by three layer DPs. Start the factor number expectation computation from the bottom. In the Section 6.3.2, the customers selecting*



(a) parameter α_0



(b) parameter α_a



(c) parameter α_d

Figure 6.4: The comparisons between empirical and expected factor numbers of CHDP with two representations: Stick and IRP. Since the factor number is parameterized by α_0, α_a and α_d.

tables in a restaurant follow a DP, and considering the expectation of DP (Ferguson, 1973), the expected number of tables in a restaurant d can be derived from

$$T_d = \sum_{n=1}^{N_d} \frac{\alpha_d}{n - 1 + \alpha_d} \tag{6.34}$$

where N_d is the number of customers in restaurant d.

A little difficult part is the second layer DPs or the expected option numbers of chefs. For the HDP, it is simply by counting all the tables in the restaurants *served by the same chef. The CHDP differs from the HDP in that the expected option number of a chef at the second layer is derived by counting all the tables served by this chef in all the restaurants. Since there is a* Cooperation *between the DPs at the second layer in CHDP, the tables in a restaurant are not served by a single chef. As shown in Eq. (6.5), the* Cooperation *has the weights* $\{\varpi_{d,a}\}$ *and* $\sum_a \varpi_{d,a} = 1$, *so the expected number of tables in restaurant* d *from chefs* a *is* $\varpi_{d,a} T_d$. *The total expected number of the tables served by chef* a *is*

$$T_a = \sum_{d}^{D} \varpi_{d,a} T_d \tag{6.35}$$

where $\varpi_{d,a}$ *is the weight of link between node* a *at the second layer and node* d *at the third layer.* T_a *can be seen as the observations of DP of chef* a, *so the expected option number of* a *is*

$$O_a = \sum_{t=1}^{T_a} \frac{\alpha_a}{t - 1 + \alpha_a} \tag{6.36}$$

Lastly, all the menu options are, in turn, seen as the observations of the global DP, so the expected factor number is

$$\mathbb{E}[K] = \sum_{o=1}^{\sum_a O_a} \frac{\alpha_0}{o - 1 + \alpha_0} \tag{6.37}$$

Theorem 5 is proven.

The above theoretical result is also empirically verified through experiments. As discussed, the expected factor numbers from the Bayesian nonparametric prior are parameterized by a set of variables. More specifically, the expected factor number of CHDP (including two representations: stick-breaking and IRP) is parameterized by α_0, α_a and α_d. In the CHDP, a cooperative hierarchical structure with size $A = 10$, $D = 20$ and $V = 50$ is firstly randomly generated. The links between nodes at three layers are also randomly set. The models (i.e., stick-breaking in Algorithm 6.1 and IRP in Algorithm 6.2) was run on this generated cooperative hierarchical structure with different values of parameters and ignoring the data likelihood, and then compared the learned empirical factor number and the theoretical factor number. The results are shown in Fig. 6.4. Three subfigures denote the three parameters of the model, respectively. From these results, it can be seen that the theoretical factor number is very close to the empirical factor number, so we have verified the result on the expected factor number. Moreover, we have demonstrated the trend of the expected factor number under the change of different parameters (parameter sensitivity). From these figures, it can be seen that the expected factor number is very sensitive to the value of α_0 but not α_a and α_d.

6.4 Mixed Gamma-negative binomial processes

This section firstly proposes a mixed Gamma-negative binomial processes model (MGNBP) while *author-paper-word* is still kept as an example to explain why this model could be used for multi-label document learning; then introduces a Gibbs sampling strategy to inference the proposed model; The significant property, i.e., expectation of topic number, from the proposed model is theoretically and empirically analyzed.

6.4.1 Model description

Since the proposed model is built on Gamma-negative binomial process, it is firstly introduced in detail followed by the propose model. Normally, the negative binomial process is used as the likelihood part in a Bayesian nonparametric

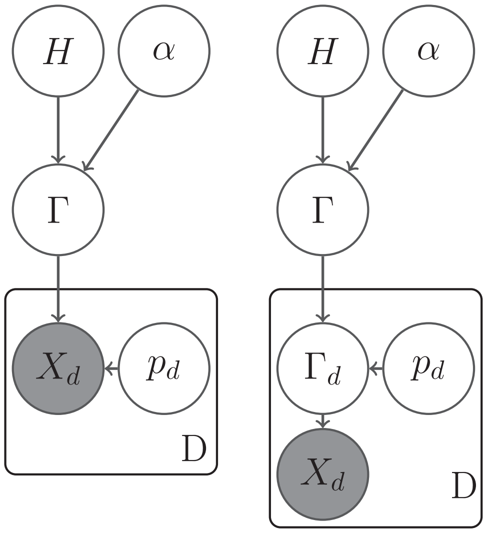


Figure 6.5: Gamma-negative binomial process topic model. The left subfigure is related to Eq. (6.38) and the right hand part is related to Eq. (6.39).

model. Analogous to a negative binomial distribution $x \sim NB(r, p)$ which has two parameters: $r > 0$ and $p \in [0, 1]$, there are two kinds of priors for a negative binomial process: one is Gamma process (Zhou and Carin, 2015) and the other is the beta process (Broderick et al., 2015). In this chapter, the Gamma process prior is used. A Gamma-negative binomial process-based topic model is proposed in (Zhou and Carin, 2015) as shown in Fig. 6.5 and can be represented as

$$\begin{aligned} \Gamma_0 &\sim GaP(c_0, H) \\ X_d &\sim NBP(p_d, \Gamma_0) \end{aligned} \tag{6.38}$$

where p_d is a real-valued parameter within $[0, 1]$ and the base measure of the negative binomial process Γ_0 is a random measure from a Gamma process. X_d is for a document, and this hierarchial form makes the documents share a same base measure Γ_0. This Gamma-negative binomial process can be (in distribution) equivalently augmented as Gamma-Gamma-Poisson process

$$\begin{aligned} \Gamma_0 &\sim GaP(c_0, H) \\ \Gamma_d &\sim GaP\left(\frac{1-p_d}{p_d}, \Gamma_0\right) \\ X_d &\sim PP(\Gamma_d) \end{aligned} \tag{6.39}$$

where $PP(\Gamma_d)$ is a Poisson process with parameter Γ_d. This augmentation is useful for the closed-form model inference algorithm design.

Consider the Gamma-negative binomial process topic model in Eqs. (6.38) and (6.39): despite its successful, this model however is fundamentally the same as the basic topic models, which are used for modeling the data of two layer hierarchy: instance-feature (i.e., document-word). Multi-label learning requires to model the data with three-layer hierarchy: label-instance-feature (i.e., author-document-word). So an intuitive idea is to add another Gamma process layer to capture the additional label (i.e, author) layer based on the Gamma-negative binomial process topic model in Eq. (6.39) analogues to the hierarchical mechanism of Hieratical Dirichlet Process (Teh, 2006)

$$\begin{aligned} \Gamma_0 &\sim GaP(c_0, H) \\ \Gamma_a &\sim GaP(c_a, \Gamma_0) \\ \Gamma_d &\sim GaP((1-p_d)/p_d, \Gamma_a^d) \\ X_d &\sim PP(\Gamma_d) \end{aligned}$$

where Γ_a is the additional layer for the label (i.e, author). This model is called a three-layer Gamma-negative binomial process topic model (3GNB), which is graphically shown in the left subfigure of Fig. 6.6.

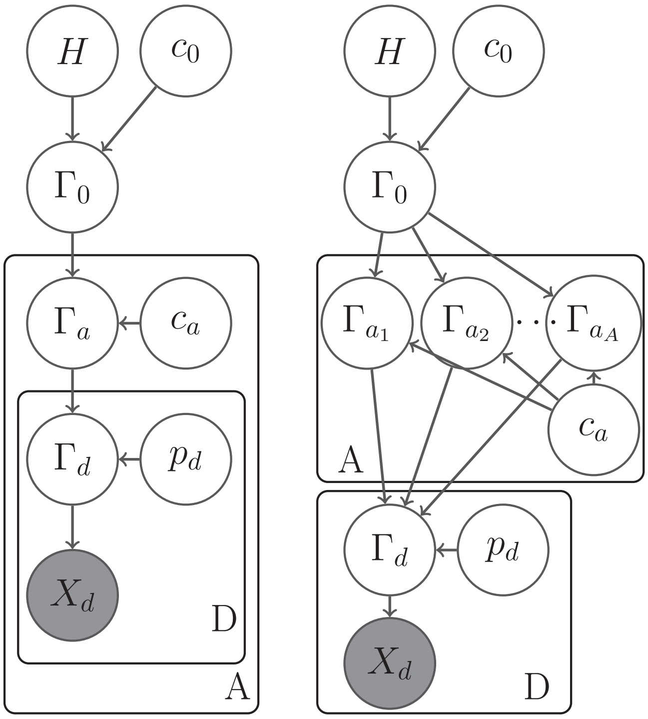


Figure 6.6: Gamma-Gamma-negative binomial process topic model (left one) and mixed Gamma-negative binomial process topic model (right one).

More specific, the global measure in the 3GNB model is

$$\Gamma_0 = \sum_{k=1}^{\infty} r_{0,k} \delta_{\theta_k} \tag{6.40}$$

where $r_{0,k}$ is the global weight of topic θ_k. This global measure defines a set of global topics $\{\theta_k\}_{k=1}^{\infty}$ shared by all documents, and $\{r_{0,k}\}_{k=1}^{\infty}$ indicates the overall "interests" of documents on topics. The number of topics can potentially be infinite and therefore justifies the infinity in the summation. However, since the data is limited, the learned topics will be also limited. Each author a is then assigned a realization of Gamma process

$$\Gamma_a = \sum_{k=1}^{\infty} r_{a,k} \delta_{\theta_k} \tag{6.41}$$

where $r_{a,k}$ is the weight of k-th topic θ_k which is inherited from the global measure Γ_0. $\{r_{a,k}\}_{k=1}^{\infty}$ can be viewed as the "interest" of author a on the topics $\{\theta_k\}_{k=1}^{\infty}$. Similar to the author, each document is also assigned a realization of Gamma process

$$\Gamma_d = \sum_{k=1}^{\infty} r_{d,k}\delta_{\theta_k} \tag{6.42}$$

where $\{r_{d,k}\}_{k=1}^{\infty}$ is the weight of "interests" of document d on the topics inherited from the global measure Γ_0 again. In the 3GNB model, the base measure for Γ_d is from its author Γ_a. It can be seen as *Inheritance*. Finally, the likelihood is a realization of Poisson process

$$X_d = \sum_{k=1}^{\infty} n_{d,k}\delta_{\theta_k} \tag{6.43}$$

where $n_{d,k}$ is the number of words in document d assigned to topic k.

When applying 3GNB to multi-label learning, there is a significant issue that each Γ_d could only have one parent Γ_a as its base measure which means that each instance is with one and only one label (i.e., a document could only have one author). In order to resolve this issue, the innovative idea is to combine all the Gamma processes of every authors of a document together by

$$\Gamma_a^d = \varpi_{a_1}^d \Gamma_{a_1} + \varpi_{a_2}^d \Gamma_{a_2} + \cdots + \varpi_{a_{A_d}}^d \Gamma_{a_{A_d}} \tag{6.44}$$

where A_d is the number of authors of document d; $\varpi_{a_1}^d$ is the weight of label a_1 on instance d; $\sum_a \varpi_a^d = 1$ (i.e., the contribution of author a_1 to document d); and Γ_a^d is the mixed prior for Γ_d. Note that the plus here is element-wise because each Γ_a is with infinite number of component. This element-wise plus operation is reasonable because the components of each Γ_a is countable and they are all with same discrete base measure Γ_0. The mixed Gamma process Γ_a^d can be seen as the *Cooperation* of all the authors of a document. To summarize, the proposed Mixed Gamma-Negative Binomial Processes-based Topic Model (MGNBP) is as follow

$$
\begin{aligned}
\Gamma_0 &\sim GaP(c_0, H) \\
\Gamma_a &\sim GaP(c_a, \Gamma_0) \\
\Gamma_a^d &= \varpi_{a_1}^d \Gamma_{a_1} + \varpi_{a_2}^d \Gamma_{a_2} + \cdots + \varpi_{a_A}^d \Gamma_{a_A} \\
\Gamma_d &\sim GaP((1-p_d)/p_d, \Gamma_a^d) \\
X_d &\sim PP(\Gamma_d)
\end{aligned}
$$

and its graphical representation is shown in right subfigure of Fig. 6.6.

6.4.2 Model inference

It is difficult to perform posterior inference under infinite mixtures, a common work-around solution in Bayesian nonparametric learning is to use a truncation method (Fox et al., 2011*a*; Blei, Griffiths and Jordan, 2010). Truncation method is widely accepted, which uses a relatively big $K^\dagger$ as the (potential) maximum number of topics. Under the truncation, the model can be expressed below as a good approximation to the infinite model

$$
\begin{aligned}
\gamma_0 &\sim Gamma(e_0, 1/f_0) \\
r_{0,k}|\gamma_0, c_0 &\sim Gamma(\gamma_0/K^\dagger, 1/c_0) \\
r_{a,k}|r_0, c_a &\sim Gamma(r_{0,k}, 1/c_a) \\
p_d &\sim beta(a_0, b_0) \\
r_{a,k}^d &= \varpi_{a_1}^d r_{a_1,k} + \varpi_{a_2}^d r_{a_2,k} + \cdots \varpi_{a_{A_d}}^d r_{a_{A_d},K^\dagger} \\
r_{d,k}|r_a, p_d &\sim Gamma(r_{a,k}^d, p_d/(1-p_d))
\end{aligned}
$$

$$
\begin{aligned}
n_{d,k} &\sim Pois(r_{d,k}) \\
N_d &= \sum_{k=1}^{K^\dagger} n_{d,k} \\
\theta_{1:K^\dagger} &\sim \frac{1}{\gamma_0} H \\
z_{d,n} &\sim Multi(r_{d,1}/\sum r_d, \cdots, r_{d,K^\dagger}/\sum r_d) \\
w_{d,n} &\sim \theta_{z_{d,n}}
\end{aligned}
$$

where $\gamma_0 = \int dH$ is the total mass of measure H and the parameters are given the appropriate priors. Here, H is a V-dimensional Dirichlet distribution, and each θ is a topic that is a V-dimensional vector.

The difficult part of the inference for this model is the mixed part Γ_a^d or r_a^d. Since r_a^d is the mixed value, it is hard to infer the posterior of r_a through its likelihood. In order to resolve this issue, the Additive Property of the negative binomial distribution is firstly introduced.

Theorem 6 *If X_i follows a negative binomial distribution with parameters r_i and p and if the various X_i are independent, then $\sum X_i$ follows a negative binomial distribution with parameters $\sum r_i$ and p.*

In MGNBP model,

$$
\begin{aligned}
r_{d,k}|\{r_a\}, p_d &\sim Gamma(r_{a,k}^d, p_d/(1-p_d)) \\
n_{d,k} &\sim Pois(r_{d,k})
\end{aligned} \tag{6.45}
$$

which are (in distribution) equal to

$$
n_{d,k} \sim NB(r_{a,k}^d, p_d) \tag{6.46}
$$

and according to Theorem 6, it is further (in distribution) equal to

$$\begin{aligned} n^a_{d,k} &\sim NB\left(\varpi^d_a \cdot r_{a,k}, p_d\right) \\ n_{d,k} &= \sum_a n^a_{d,k} \end{aligned} \tag{6.47}$$

where $\{n^a_{d,k}\}$ are independent with each others.

$n_{d,k}$ the number of words assigned to topic k in document d have been split into a number A_d of independent variables $\{n^a_{d,k}\}$. Here, $n^a_{d,k}$ denotes the number of words assigned to topic k from author a in document d. From Eq. (6.47), it can be seen that if the likelihood part of the r_a is known, the r_a can be updated using n^a_d. Introducing the auxiliary variables $\{n^a_{d,k}\}$ helps us resolve the difficult inference problem brought by the mixed Gamma process. Note that the independence between the elements of $\{n^a_{d,k}\}$ is very important, which facilitates us update each $n^a_{d,k}$ independently.

According to the relationship between the negative binomial distribution and the Gamma-Poisson distribution, for each $n^a_{d,k}$,

$$n^a_{d,k} \sim NB(\varpi^d_a r_{a,k}, p_d) \Longrightarrow \quad r^a_{d,k} \sim Gamma(\varpi^d_a r_{a,k}, \frac{p_d}{1-p_d}), n^a_{d,k} \sim Pois(r^a_{d,k}) \tag{6.48}$$

It needs to highlight that $r^a_{d,k}$ is different from $r^d_{a,k}$; $r^d_{a,k}$ is the mixed Gamma process of multiple author Gamma processes Γ_a of Gamma process Γ_d of document d and $r^a_{d,k}$ is the interest of document d on topic k inherited from author a.

Due to the non-conjugacy of Gamma distribution and negative binomial distribution, it is difficult to update r_a with a Gamma prior. In order to make the inference with only close-formed conditional distributions, the following result on the negative binomial process is used

Theorem 7 *(Zhou and Carin, 2015) If X follows a negative binomial distribution $X \sim NB(r,p)$ with parameters r and p, then X can also be generated from a compound Poisson distribution as*

$$X = \sum_{t=1}^{l} u_t, u_t \overset{i.i.d}{\sim} Log(p),\ l \sim poiss\left(-rln(1-p)\right) \tag{6.49}$$

where $Log()$ is a Logarithmic distribution. Furthermore, this Poisson-logarithmic bivariate count distribution, $p(X,l)$, can be expressed as

$$X \sim NB(r,p),\ \ l \sim CRT(X,r) \tag{6.50}$$

where $CRT()$ denotes a Chinese Restaurant distribution, and its definition and sampling can be found in (Zhou and Carin, 2015).

With Theorem 7, the Eq. (6.48) is also equal to

$$\begin{aligned} n^a_{d,k} \sim NB(\varpi^d_a \cdot r_{a,k}, p_d) &\Longrightarrow n^a_{d,k} \sim \sum_{1}^{l^a_{d,k}} log(p_d), l^a_{d,k} \sim Pois(-\varpi^d_a r_{a,k} ln(1-p_d)) \\ &\Longrightarrow l^a_{d,k} \sim CRT(n^a_{d,k}, \varpi^d_a r_{a,k}), n^a_{d,k} \sim NB(\varpi^d_a r_{a,k}, p_d) \end{aligned} \tag{6.51}$$

Finally, all $n^a_{d,k}$ can be updated by

$$\begin{aligned} (n^{a_1}_{d,k_1}, \cdots, n^{a_A}_{d,K}) &\sim Mult(n_d, \frac{\varpi^d_{a_1} r^{a_1}_{d,k_1}}{r_d}, \cdots, \frac{\varpi^d_{a_{A_d}} r^{a_A}_{d,K}}{r_d}) \\ r_d &= \sum_a \sum_k \varpi^d_a \cdot r^a_{d,k} \end{aligned} \tag{6.52}$$

and for each word n in a document d, assign it to a topic k and author a by

$$\begin{aligned} p(z_{d,n} = k, \hat{z}_{d,n} = a) \propto \frac{\varpi_a^d r_{d,k}^a}{r_d} \\ n_{d,k} = \sum_n \delta(z_{d,n} = k) \\ n_{a,k} = \sum_d \sum_n \delta(z_{d,n} = k \ \& \ \hat{z}_{d,n} = a) \end{aligned} \tag{6.53}$$

where $z_{d,n}$ is the topic index assigned to word n in document d.

With these changes of variables, the original model is re-formulated as

$$\begin{aligned} \gamma_0 &\sim Gamma(e_0, 1/f_0) \\ r_{0,k}|\gamma_0, c_0 &\sim Gamma(\gamma_0/K^\dagger, 1/c_0) \\ p_d &\sim beta(a_{d,0}, b_{d,0}) \\ r_{a,k}|r_0, c_a &\sim Gamma(r_{0,k}, 1/c_a) \\ r_{a,k}^d &= \varpi_{a_1}^d r_{a_1,k} + \varpi_{a_2}^d r_{a_2,k} + \cdots \varpi_{a_{A_d}}^d r_{a_{A_d},K^\dagger} \\ r_{d,k}|r_a, p_d &\sim Gamma(r_{a,k}^d, p_d/(1-p_d)) \\ r_{d,k}^a &\sim Gamma(\varpi_a^d r_{a,k}, p_d/(1-p_d)), \quad a \in A^d \\ z_{d,n}^a &\sim Discrete(\frac{\varpi_a^d r_{d,k}^a}{r_d}, \cdots) \\ n_{d,k} &= \sum_n \delta(z_{d,n} = k) \\ n_{a,k} &= \sum_d \sum_n \delta(z_{d,n} = k \ \& \ \hat{z}_{d,n} = a) \\ n_{d,k}^a &= \sum_n \delta(z_{d,n} = k \ \& \ \hat{z}_{d,n} = a) \\ N_d &= \sum_n \sum_a z_{d,n}^a \end{aligned}$$

where A^d is the set of associated authors of document d.

In the following, a Gibbs sampling algorithm is designed for the posterior inference and all the conditional distributions are listed below.

Sampling z

$$p(z_{d,n} = k, \hat{z}_{d,n} = a | \cdots) \propto \theta_{k,n} \cdot \varpi_a^d r_{d,k}^a \tag{6.54}$$

Sampling r_d^a

$$p(r_{d,k}^a | \cdots) \propto Gamma(\varpi_a^d r_{a,k} + n_{d,k}^a, p_d) \tag{6.55}$$

Sampling l_d^a

$$p(l_{d,k}^a | \cdots) \propto CRT\left(n_{d,k}^a, \varpi_a^d r_{a,k}\right) \tag{6.56}$$

Sampling p_d

$$\begin{aligned} r_{a,k}^d &= \varpi_{a_1}^d r_{a_1,k} + \varpi_{a_2}^d r_{a_2,k} + \cdots \\ p(p_d | \cdots) &\propto Beta\left(a_0 + \sum_k n_{d,k}, b_0 + \sum_k r_{a,k}^d\right) \\ p(r_{d,k} | \cdots) &\propto Gamma(r_{a,k}^d + n_{d,k}, p_d) \end{aligned} \tag{6.57}$$

Sampling r_a

$$p(r_{a,k} | \cdots) \propto Gamma\left(r_{0,k} + \sum_{d \text{ with } a} l_{d,k}^a, \frac{1}{c_a - \sum_{d \text{ with } a} \varpi_a^d \cdot \ln(1 - p_d)}\right) \tag{6.58}$$

Sampling l_a

$$p(l_{a,k} | \cdots) \propto CRT\left(\sum_{d \text{ with } a} l_{d,k}^a, r_{0,k}\right) \tag{6.59}$$

Sampling $r_{0,k}$

$$p(r_{0,k}|\cdots) \propto Gamma\left(\gamma_0/K^\dagger + \sum_a l_{a,k}, \frac{1}{c_0 - \sum_a \ln(1-p_a)}\right) \tag{6.60}$$

where

$$p_a = \frac{-\sum_{d \text{ with } a} \varpi_a^d ln(1-p_d)}{c_a - \sum_{d \text{ with } a} \varpi_a^d ln(1-p_d)} \tag{6.61}$$

Sampling l'_k

$$p(l'_k|\cdots) \propto CRT\left(\sum_a l_{a,k}, \gamma_0/K^\dagger\right) \tag{6.62}$$

Sampling γ_0

$$p(\gamma_0|\cdots) \propto Gamma\left(e_0 + \sum_k l'_k, \frac{1}{f_0 - \ln(1-p')}\right) \tag{6.63}$$

where

$$p' = \frac{-\sum_a ln(1-p_a)}{c_0 - \sum_a ln(1-p_a)} \tag{6.64}$$

Sampling θ_k

$$p(\theta_k|\cdots) \propto H(\theta_k) \prod_d \theta_{z_{d,n}=k,n} \tag{6.65}$$

It can be seen from these conditional distributions that all of them are closed-form which is very easy to update and implement. The whole procedure is summarized in Algorithm 6.3. After obtaining the samples of the posterior $p(\theta, r_a, r_d, r_0, z^a_{d,n}, p_d, \gamma_0, n^a_{d,k}|\cdots)$ of latent variables and removing the burn-in stage, the algorithm firstly identifies the topic number with largest frequency as the K_{real}, and then find the sample with largest likelihood and $K = K_{real}$ from these samples. The output of Gibbs sampler are the latent variables θ, r_a and r_d in this sample.

Algorithm 6.3: Gibbs Sampler for MGNBP

```
Input: D, A, N, AD, DN
Output: K_real, {θ}, {r_a}, {r_d}
initialization;
while iter ≤ max_iter do
    for d = 1; d ≤ D do
        for n = 1; n ≤ N_d do
            Update z_{d,n} and ẑ_{d,n} by Eq. (6.54);
        for a = 1; a ≤ A_d do
            Update r^a_{d,k} by Eq. (6.55);
            Update l^a_{d,k} by Eq. (6.56);
        Update r_{d,k} and p_d by Eq. (6.57);
    for a = 1; a ≤ A do
        Update r_{a,k} by Eq. (6.58);
        Update l_{a,k} by Eq. (6.59);
    Update r_{0,k} by Eq. (6.60);
    Update l'_k by Eq. (6.62);
    Update γ_0 by Eq. (6.63);
    Update θ by Eq. (6.65);
    iter++;
Identify K_real;
Select the sample with largest likelihood and K = K_real;
return {θ}, {r_a}, {r_d};
```

6.4.3 Model analysis

A distinguishing characteristic of Bayesian nonparametric model is that the number of the factors/topics to be learned is not specified a priori. Roughly speaking, Bayesian nonparametric model could be simply seen as a prior for the this number. Conditioned on the observed data, the model could determine how many factors/topics are needed. It would be interesting to investigate the prior expectation of the factors/topics number under the defined model. The following result is given

Theorem 8 *Given D instances, A labels, and their mapping AD, the expected* *factor number from the MGNBP is*

$$\int_{r_0} \left(1 - \prod_a \left[\frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}\right]^{r_0}\right) \cdot \frac{\gamma_0 \cdot \exp(-c_0 \cdot r_0)}{r_0} \cdot \mathrm{d}r_0 \tag{6.66}$$

and when a truncation level $K^\dagger$ *is applied, the expected factor number is*

$$K^\dagger \left(1 - \left[\frac{c_0}{c_0 - \sum_a \log \frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}}\right]^{\frac{\gamma_0}{K^\dagger}}\right) \tag{6.67}$$

where γ_0*,* c_0*,* c_a *and* p_d *are four parameters of the MGNBP.*

Proof 7 *It needs to firstly introduce the following theorem of a completely random measure*

Theorem 9 *(Kingman, 1992)* ***Campbell's Theorem*** *Let* Π *be a Poisson process on* Θ *with mean measure* μ*, and let* $f : \Theta \to \mathbb{R}$ *be measurable. Then the sum*

$$\sum = \sum_{Y \in \Pi} f(Y) \tag{6.68}$$

is absolutely convergent with probability if and only if

$$\int_\Theta \min(|f(y)|, 1)\mu(\mathrm{d}y) < \infty \tag{6.69}$$

If this condition holds, the expectation

$$\mathbb{E}[\sum] = \int_\Theta f(y)\mu(\mathrm{d}y) \tag{6.70}$$

exists if and only if the integral converges.

Since the proposed MGNBP is a completely random measure, the above theorem can be utilized to compute the expectation of sum of its variables. Define a *random variable*

$$\mathbb{X}_k = \mathbb{1}(\sum_{d=1}^{D} \sum_{AD[a,d]=1} C^a_{d,k} > 0) \tag{6.71}$$

which equals to 1 if the factor k is used; 0, otherwise. The expected factor number is $\mathbb{E}[\sum_k \mathbb{X}_k]$. *Then, according to the Theorem 9,*

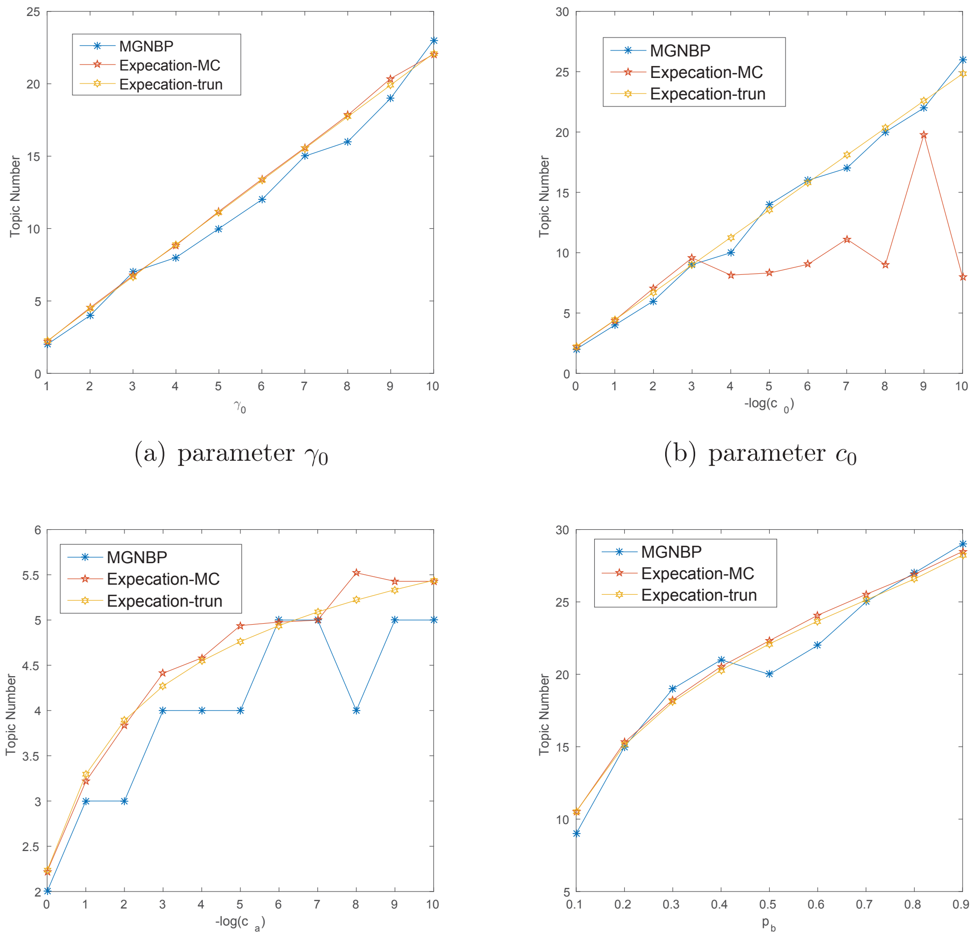
$$\begin{aligned}
&\mathbb{E}[\sum_k \mathbb{X}_k] \\
&= \mathbb{E}\left[\mathbb{E}\left[\sum_k \mathbb{X}_k | r_{0,k}\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\mathbb{E}\left[\sum_k \mathbb{X}_k | \{r_{a,k}\}\right] | r_{0,k}\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\sum_k \mathbb{E}\left[\mathbb{X}_k | \{r_{a,k}\}\right] | r_{0,k}\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\sum_k \left(1 - \prod_{d=1}^{D} \prod_{a:AD[a,d]>0} (1-p_b)^{\varpi_a^d r_{a,k}}\right) | r_{0,k}\right]\right] \\
&= \mathbb{E}\left[\sum_k \mathbb{E}\left[\left(1 - \prod_{d=1}^{D} \prod_{a:AD[a,d]>0} (1-p_d)^{\varpi_a^d r_{a,k}}\right) | r_{0,k}\right]\right] \\
&= \mathbb{E}\Big[\sum_k \int_{r_{1,k}} \cdots \int_{r_{A,k}} \left(1 - \prod_{d=1}^{D} \prod_{a:AD[a,d]>0} (1-p_d)^{\varpi_a^d r_{a,k}}\right) \\
&\qquad \cdot \left\{\prod_{a=1}^{A} \frac{{c_a}^{r_{0,k}}}{\Gamma(r_{0,k})} {r_{a,k}}^{r_{0,k}-1} \exp(-c_a \cdot r_{a,k})\right\} \mathrm{d}r_{1,k} \cdots \mathrm{d}r_{A,k}\Big] \\
&= \mathbb{E}\Big[\sum_k \left(1 - \prod_a \left[\frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}\right]^{r_{0,k}}\right)\Big] \\
&= \int_{r_0} \left(1 - \prod_a \left[\frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}\right]^{r_0}\right) \cdot \nu_{GaP}(r_0) \ \cdot \ \mathrm{d}r_0 \\
&= \int_{r_0} \left(1 - \prod_a \left[\frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}\right]^{r_0}\right) \cdot \frac{\gamma_0 \cdot \exp(-c_0 \cdot r_0)}{r_0} \cdot \mathrm{d}r_0
\end{aligned}$$

This integral cannot be easily solved. An approximate method for this integral computation is Monte Carlo. If applying a truncation level $K^\dagger$, the expectation is

$$
\begin{aligned}
\mathbb{E}[\sum_k^{K^\dagger} \mathbb{X}_k] = & \int_{r_{0,1}\cdots r_{0,K^\dagger}} \sum_k^{K^\dagger} \left(1 - \prod_a \left[\frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}\right]^{r_{0,k}}\right) \\
& \cdot \prod_{k=1}^{K^\dagger} \frac{{c_0}^{\gamma_0/K^\dagger}}{\Gamma(\gamma_0/K^\dagger)} {r_{0,k}}^{\gamma_0/K^\dagger - 1} \exp(-c_0 r_{0,k}) \mathrm{d}r_{0,1} \cdots \mathrm{d}r_{0,K^\dagger} \\
= & \sum_k \left\{ 1 - \left[\frac{c_0}{c_0 - \sum_a \log \frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}}\right]^{\frac{\gamma_0}{K^\dagger}} \right\} \\
= & K^\dagger \left(1 - \left[\frac{c_0}{c_0 - \sum_a \log \frac{c_a}{c_a - \sum_{d:AD[a,d]>0} \varpi_a^d \ln(1-p_d)}}\right]^{\frac{\gamma_0}{K^\dagger}}\right)
\end{aligned}
$$

The theorem is proofed.

The above theoretical result is also supported by simulation results, summarized in Fig. 6.7. At first, set $A = 10$, $D = 20$, and the mapping relations between labels and instances are also randomly generated. Simulate the model with the above setting and different values of parameters, and then compare the empirical factor number and the theoretical factor number from Theorem 8. The default values of them are: $\gamma_0 = 1$, $c_0 = 1$, $c_a = 1$ and $p_d = 0.5$. When investigating one parameter, the other three will be fixed as the default values. Four subfigures in Fig. 6.7 denote the changing of factor number as the changing of four parameters of the model, respectively. In each subfigure, *Expectation-MC* denotes the Monte Carlo approximation of the expectation of the factor number from Eq. (6.66); *Expectation-trun* denotes the truncation-based approximation of the expectation of the factor number from Eq. (6.67) (Note that the implementation of the MGNBP is based on the truncation $K^\dagger = 1000$); the x-axes of c_0 and c_a are in negative (base-10) log space. From these results, it can be seen



(a) parameter γ_0 (b) parameter c_0

(c) parameter c_a (d) parameter p_d

Figure 6.7: The comparisons between expected and empirical factor numbers of MGNBP under different parameters: γ_0, c_0, c_a and p_d. Note that the x-axes of c_0 and c_a are in negative (base-10) log space.

that the theoretical factor number is very close to the empirical factor number, so this verified the results on the expected factor number of the MGNBP. The trends of the empirical and expected factor number with parameters γ_0 and p_d are extremely close with each others. For the parameter c_a, the trends are also close; for the parameter c_0, *Expectation-MC* is a little away from the others as the increasing of the value of c_0. These subfigures do not only verify the above theoretical result, they also show the sensitivity of the model to the parameters.

6.5 Experimental evaluation

Two experiments in this section to demonstrate the proposed models on cooperatively hierarchical structure modeling using the real-world tasks. The public datasets used for the experimental evaluation are:

- **NIPS papers**[1] This dataset contains papers from the NIPS conferences between 1987 and 1999. This dataset is a structure: *author-paper-word*. It contains 1,740 papers with 2,037 authors, a total of 2,301,375 word tokens and a vocabulary size of 13,649 unique words. More description can be found in (Steyvers et al., 2004);

- **Clinical free text labeling**[2]. They are primary data about patients. This dataset is a structure: *label-instance-feature*. There are 45 labels (like ICD-9-CM codes) and 645 (training) / 333 (testing) data with 1,449 features. More description can be found in (Pestian et al., 2007).

Note that three datasets for the three tasks all involve a cooperative hierarchical structure which cannot be modeled by existing Bayesian non- parametric models, like HDP. Next, each task is introduced in more detail, including the aim, comparative models, evaluation metrics, and the results analysis. the results analysis.

6.5.1 Author-topic model task

The first task for the proposed models for cooperative hierarchical structures is to model an *author-document-word* structure. The aim of this task is to

[1] http://www.datalab.uci.edu/author-topic/NIPs.htm
[2] http //mulan.sourceforge.net/datasets-mlc.html

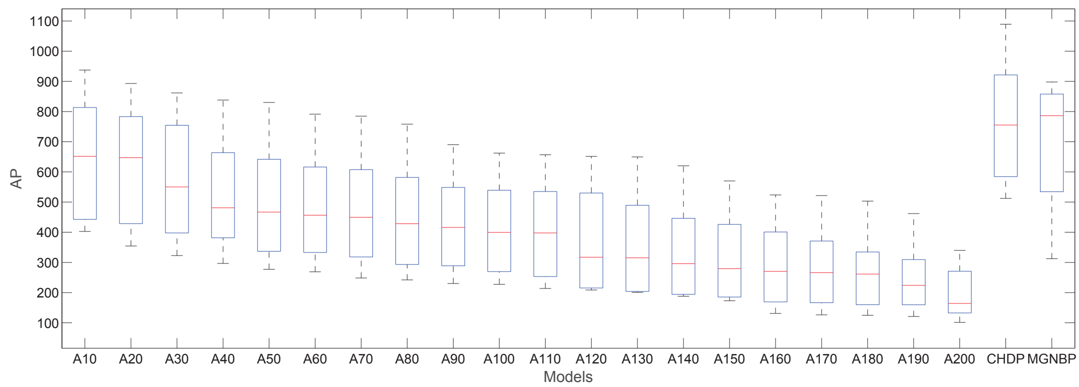


Figure 6.8: The comparisons between the ATM (fixed dimensional model) with CHDP and MGNBP (nonparametric models) on the author prediction. 'A$\mathcal{X}$' denotes the ATM with $\mathcal{X}$ number of topics. Each box expresses the statistics of results from the corresponding model on 5-fold cross-validation. The larger the value of AP, the better the performance.

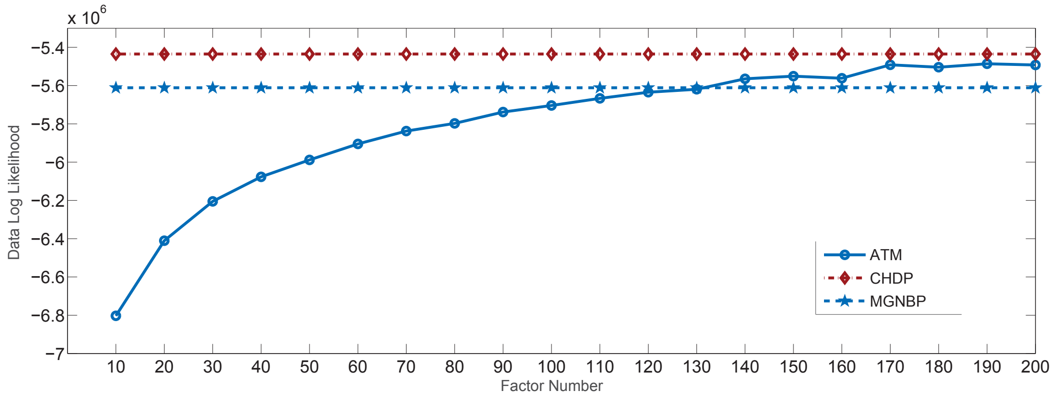


Figure 6.9: The comparisons between the ATM (fixed dimensional model) with CHDP and MGNBP (nonparametric models) on the data likelihood. Since CHDP and MGNBP do not need the prefixed topic number, two lines are plotted in this figure.

discover the hidden topics from the **NIPS papers** and, simultaneously, the authors' interests in these topics. This task can be further applied to real-world applications, such as 1) detecting the Most and Least Surprising Papers for an Author, 2) an author-topic browser; 3) author disambiguation; and so on. The selected comparative model for this task is author topic model (ATM) (Rosen-Zvi et al., 2004; Steyvers et al., 2004; Rosen-Zvi et al., 2010), which is based on the fixed dimensional probability distributions. Therefore, the topic number needs to be fixed when using ATM for this task, but the proposed models in this chapter do not suffer this problem. The setting for this task evaluation is as follows: keep 20% documents as the test dataset and consider the remaining 80% documents as the training dataset. After learning the proposed models on the training dataset, predict the authors of test documents. Two evaluation metrics were used for the qualitative comparisons.

Author-prediction accuracy:

$$AP = \sum_{d} \sum_{a} < \pi_d, \pi_a > \tag{6.72}$$

where $<, >$ denotes the cosine similarity between two vectors; π_a represents the interest of author a on the topics

$$\pi_{a,k} = \sum_{o}^{O_a} \pi_{a,o} \mathbb{1}(z_{a,o} = k), \;\; k \in [1, K] \tag{6.73}$$

and π_d represents the interest of paper d on the topics

$$\pi_{d,k} = \sum_{n}^{N_d} \theta_{k,v_n}, \quad k \in [1, K] \tag{6.74}$$

where n denotes a word in paper d, v_n is the word name of word n in paper d. The larger the value of AP, the better the performance.

Data (log) likelihood:

$$\mathcal{L} = \sum_{d}^{D_{training}} \sum_{n}^{N_d} \log \sum_{k}^{K} \theta_{k,v_n} \tag{6.75}$$

where N_d is the number of words in paper d and $D_{training}$ is the number of training papers in the dataset. *Data (log) likelihood* is an evaluation of the model fitting the training dataset. If the learned topics are better, *data (log) likelihood* should be larger.

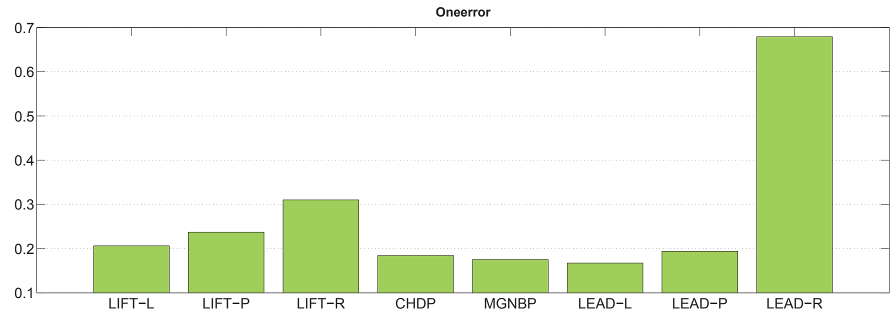
The results have been shown in Fig. 6.8. Since ATM needs the number of topics to be fixed in advance, the 20 candidates $K \in \{i : i = j \times 10, j = [1, 20]\}$ were evaluated. In Fig. 6.8, 'A$\mathcal{X}$' denotes the ATM with $\mathcal{X}$ number of topics, and CHDP is implemented by the stick-breaking representation in Section 6.3.1 and Algorithm 6.1. For **NIPS papers** dataset, 5-fold cross validation is conducted: the entire dataset was separated into 5 parts and one of them was used as the test data for each time. Each box in the Fig. 6.8 denotes the statistics of results from the corresponding model. As shown in Fig. 6.8, the performance of the ATM on the *author prediction* tended to worsen as the number of topics increased. The proposed CHDP and MGNBP performed better than ATM with different number of topics in general. More specifically, the median of MGNBP is slightly better than the one of CHDP, but CHDP is more reliable than MGNBP (considering the minimum and the maximum values). Except the *author prediction*, the data likelihood of different models are also demonstrated in Fig. 6.9, which is an evaluation of the probability of the data under a given model, $P(data|model)$. It can be seen that, as opposed to the performance on *author prediction*, the *data likelihood* increased with an increasing topic number. Since the proposed CHDP and MGNBP do not need the prefixed topic number, there are two lines representing their data likelihoods in the Figure. In this experiment, the topic numbers between $[10, 200]$ are evaluated. From the Fig. 6.9, the data likelihood from ATM exceeded the one from MGNBP at around number 130, but the difference between their likelihoods beyond number 130 was not very large. Within the scope of $[10, 200]$, the data likelihood from CHDP was always better than ATM, but it can be seen that the data likelihood from ATM will exceed CHDP as the further increasing of the topic number. To summarize, there is a, somewhat lax, contradiction between the ability to fit the training data and the ability to predict the test data. The CHDP and MGNBP balanced two abilities well.

6.5.2 Clinical free text labeling task

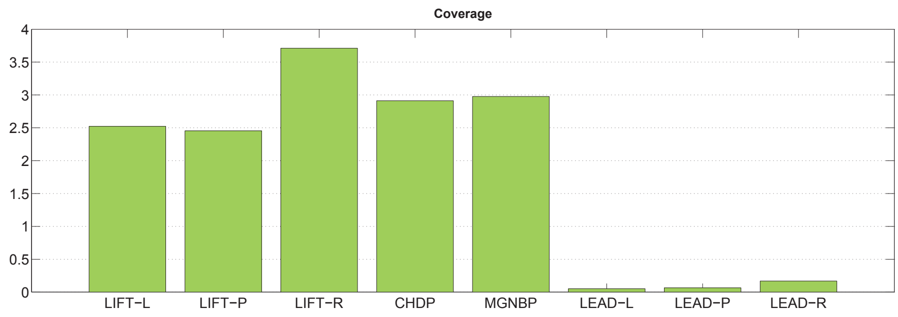
The second task for the proposed models is clinical free text labeling. The aim of this task is to automatically assign the labels to the data. Take the example of a **Clinical free text labeling** dataset. Automatic and accurate label assignment for the text can save an enormous amount of time and cost compared with to manual labor. The comparative models for this task are LEAD (Zhang and Zhang, 2010) and LIFT (Zhang and Wu, 2015; Zhang, 2011)[3], which are both deterministic models based on support vector machine (SVM). Comparing with them, the proposed models are both generative models which normally have a better generalizing ability on unseen data. The evaluation metrics used for the performance comparison on this task are: *Oneerror*, *Coverage*, *Rankingloss*, *Avgprecision*, and their definitions can be found in (Madjarov et al., 2012). For *Avgprecision*, the larger the value, the better the performance; For *Oneerror*, *Coverage* and *Rankingloss*, the smaller the value, the better the performance. These metrics are all ranking-based. They can rank all the labels in different multi-label classification models for each data according the possibility of the data with each label. The proposed models in this chapter can also rank the labels for the test data according to their interests in the hidden topics by

$$R(l, x_i) =< \vec{v_l}, \vec{v_i} > \tag{6.76}$$

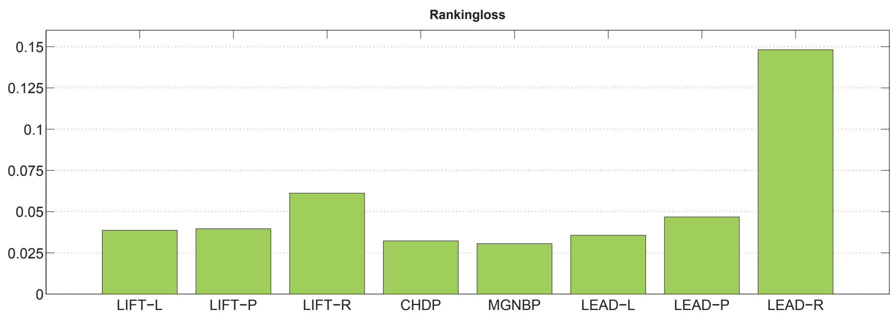
[3]Implementations are both from: http://cse.seu.edu.cn/people/zhangml/Resources.htm



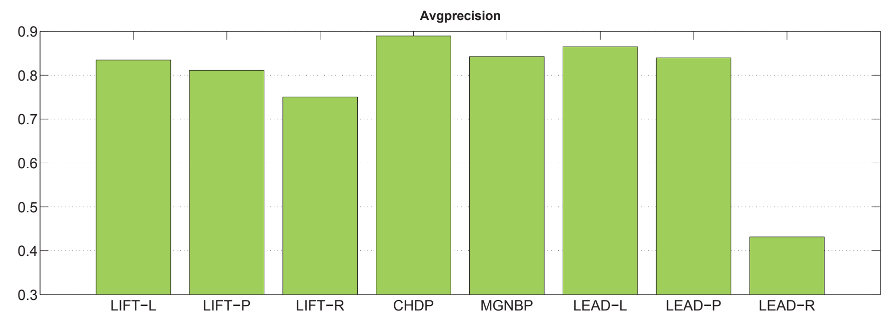
(a) *Oneerror* (The smaller the value, the better the performance.)



(b) *Coverage* (The smaller the value, the better the performance.)



(c) *Rankingloss* (The smaller the value, the better the performance.)



(d) *Avgprecision* (The larger the value, the better the performance.)

Figure 6.10: The comparisons between LIFT, LEAD, CHDP and MGNBP on the multi-label classification. Each subfigure demonstrates the results on a corresponding metric.

where x_i is i-th test data, l denotes a label, $\overrightarrow{v_l}$ is a K dimensional vector that denotes the interest of label l on the K topics can be similarly evaluated by Eq. (6.73), $\overrightarrow{v_i}$ is also a K dimensional vector that denotes the interest of data x_i on the K topics can be similarly evaluated by Eq. (6.74). The larger the value of $R(l, x_i)$, the larger possibility of x_i with label l. Therefore, labels for each data can be ranked according to $R(l, x_i)$.

The results are shown in Fig. 6.10. Since the LEAD and LIFT are SVM-based models, different implementations with different kernel functions have been compared. 'LIFT-L' denotes LIFT with Linear kernel function; 'LIFT-P' denotes LIFT with Polynomial kernel function (the degree is set as 3); 'LIFT-R' denotes LIFT with Radial basis function kernel function. In the Fig. 6.10, CHDP and MGNBP achieved best performances on *Oneerror* and *Rankingloss*, and they also achieved a comparative performance on *Avgprecision*. For the *Coverage*, LEAD achieved the best performances, while CHDP and MGNBP only surpassed the worst LIFT-R. Four metrics each have their own preferences on the classification evaluation. *Oneerror* and *Coverage* prefer 'variance', and *Rankingloss* and *Avgprecision* prefer 'mean', so the proposed models, i.e., CHDP and MGNBP, performed better when evaluated against the 'mean', but the 'variance' was not larger than LIFT and LEAD. This may be because that the proposed models are based on MCMC, so each run is a sample from the real model posterior distribution, which has a larger variance during the sampling, although the variance has already been decreased by incorporating the data. To summarize, CHDP and MGNBP had better performance than LIFT and LEAD in the clinical free text labeling task, especially considering the generative model nature of CHDP and MGNBP.

6.6 Summary

Hierarchical structures are a significant data structure adopted by different areas for the data modeling, so it has attracted many attention of many research laboratories, such as multi-label document learning. This chapter has presented two Bayesian nonparametric models, i.e., CHDP and MGNBP, for more general hierarchical structures: cooperative hierarchical structures. Two models are based on two random measure operations which have been specifically designed for modeling the cooperative hierarchical structures: Inheritance and Cooperation. In order to resolve the issue brought about by Inheritance and Cooperation in CHDP and MGNBP, Markov Chain Monte Carlo inference algorithms have been designed through introducing properly designed additional auxiliary variables. Experiments on two multi-label document learning tasks with different public datasets have shown the ability of the proposed models on modeling cooperative hierarchical structure.

Chapter 7

Conclusion and Further Study

This chapter concludes the entire thesis and provides some further research directions for this topic.

7.1 Conclusions

Bayesian nonparametric learning has attracted much attention in the area of statistical machine learning due to its powerful modeling ability and demand for less prerequisites over the past decades - especially for text mining. Even though Bayesian nonparametric learning has gained considerable attention and undergone rapid development, its ability to model or capture complex text data, such as document-word co-clustering, document networks, multi-label documents, and so on, is still lacking. Yet, text is one of the most natural and easy ways to express human knowledge and opinion and holds an irreplaceable position in modern life. Thus, modeling and analysis of complex text is of great significance to corporations, organisations and individuals, and research on Bayesian nonparametric learning for text mining has great theoretical and practical relevance. The findings of this study are summarised as follows:

1. *The development of three Bayesian nonparametric sparse nonnegative matrix factorisation models for document-word co-clustering.* (To achieve OBJECTIVE 1)

   A Bayesian nonparametric NMF framework based on dependent Indian buffet processes (dIBP) has been proposed to remove the assumption of NMF for co-clustering tasks where the dimension of the factors is known in advance. First, a model is built by implementing a framework using GP-based dIBP, which successfully removes the assumption, but suffers from larger model complexity and less flexibility. Then, two new dIBPs have been proposed through a bivariate Beta distribution and a copula. The advantages of the models based on new dIBPs are: 1) They have simpler model structures than models with GP-based dIBP; 2) The correlation between data could be directly learned and can be seen as a measurement of the focus degree of the hidden factors or topics. Lastly, three inference algorithms have been designed for the proposed three models, respectively. Experiments on synthetic and real-world datasets demonstrates the ability of the proposed models to perform NMF without predefining the dimension number. It also more correlation flexibility when compared to GP-based dIBP. The bivariate Beta distribution-based model and copula-based model both achieve comparative performance on document-word co-clustering.

2. *The development of a Bayesian nonparametric deep topic model for the topic hierarchy learning.* (To achieve OBJECTIVE 2)

   A Bayesian nonparametric deep topic model has been proposed to automatically learn a topic hierarchy from any document collection. Based on renowned hierarchical Dirichlet processes (HDP), a new probability measure has been proposed, which not only inherits the mechanisms of linking the topics at different layers but also breaks the topic sharing strategy of the original HDP. With the help of this new probability measure, a topic hierarchy can be learned. Additionally, a slice sampling method without truncation approximation has been designed. Experimental results show that the topics in higher layers are more general than the topics in lower layers, which have a better ability to predict new documents. Conversely, the topics in the lower layers are more specific than the topics in higher layers, which have a better ability to explain the training documents.

3. *The development of a Bayesian nonparametric relational topic model for document network learning.* (To achieve OBJECTIVE 3)

   A Bayesian nonparametric relational topic model has been presented in which stochastic processes are adopted to replace the fixed-dimensional probability distributions used by existing relational topic models and necessitates a pre-defined the number of topics. Introducing stochastic processes, however, leads to difficulties with model construction and inference, and this study has therefore presented a thinned gamma process-based model and has also presented truncated Gibbs and slice sampling algorithms for the proposed model. Experiments on both synthetic and real-world datasets has demonstrated the proposed methodąŕs ability to infer the hidden topics and their number.

4. *The development of Bayesian nonparametric cooperative hierarchical structure models for multi-label document learning.* (To achieve OBJECTIVE 4)

   Considering the problems of existing researches on hierarchical structures, i.e., that the number of hidden factors needs to be predefined and only designed for non-cooperative hierarchical structures, two measure operations have been defined to model the cooperative hierarchical structures: inheritance and cooperation. Two different models have been proposed, each of which exemplify the two measure operations. One is the cooperative hierarchical Dirichlet process (CHDP) with a Dirichlet process as its basic ingredient. The other is a mixed gamma-negative binomial process (MGNBP) with a gamma process as its basic ingredient. In order to resolve the issues introduced by inheritance and cooperation, three different Markov chain Monte Carlo inference algorithms have been designed through introducing properly designed auxiliary variables. Experiments on various tasks with different datasets have shown the efficiency of the proposed models, and has demonstrated wide practical value in a range of fields.

7.2 Further study

One possible future study for this work pertains to efficiency, because current Gibbs sampling-based inference lacks efficiency when dealing with big data, although it has the merit in that it is theoretically capable of reaching the exact posterior distribution. This thesis identifies the following directions as future work:

- *Variational Inference* A mean-field variational method, in which the variational distributions (normally fully-factorised) are used to approximate the true posterior distribution by adjusting their distribution parameters (Hoffman et al., 2013). This approach would transfer the sampling problem into an optimisation problem, so the inference speed could be accelerated. Developing a set of variational inference algorithms for the proposed models in this thesis would therefore, be interesting for future work.
- *Parallel Implementation* Another strategy to accelerate inference speed uses the help of a parallel machine (Gonzalez et al., 2011). Current sampling algorithms would be carefully redesigned and assigned to multi-threads (multi-core

parallel machines) or multi-machines (distributed parallel machines). Theoretical support could also be distributed along with the task assignment of the original problem.

REFERENCES

Aggarwal, C. C. and Zhai, C. (2012), *Mining Text Data*, Springer US, Boston, MA, chapter A survey of text clustering algorithms, pp. 77–128.

Airoldi, E. M., Blei, D. M., Fienberg, S. E. and Xing, E. P. (2008), 'Mixed membership stochastic blockmodels', *Journal of Machine Learning Research* **9**, 1981–2014.

Aldous, D. J. (1985), *Exchangeability and Related Topics*, Springer.

Alghamdi, R. and Alfalqi, K. (2015), 'A survey of topic modeling in text mining', *International Journal of Advanced Computer Science and Applications* **6**(1), 147–153.

Andrieu, C., de Freitas, N., Doucet, A. and Jordan, M. I. (2003), 'An introduction to mcmc for machine learning', *Machine Learning* **50**(1), 5–43.

Archambeau, C., Lakshminarayanan, B. and Bouchard, G. (2015), 'Latent ibp compound dirichlet allocation', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 321–333.

Balog, M. and Teh, Y. W. (2015), 'The mondrian process for machine learning', *arXiv preprint arXiv:1507.05181* .

Bao, S., Xu, S., Zhang, L., Yan, R., Su, Z., Han, D. and Yu, Y. (2012), 'Mining social emotions from affective text', *IEEE Transactions on Knowledge and Data Engineering* **24**(9), 1658–1670.

Beare, B. K. (2010), 'Copulas and temporal dependence', *Econometrica* **78**(1), 395–410.

Bichot, C.-E. (2010), 'Co-clustering documents and words by minimizing the normalized cut objective function', *Journal of Mathematical Modelling and Algorithms* **9**(2), 131–147.

Blei, D. M. (2004), Probabilistic Models of Text and Images, PhD thesis, Berkeley, CA, USA. AAI3183785.

Blei, D. M. (2012), 'Probabilistic topic models', *Communications of the ACM* **55**(4), 77–84.

Blei, D. M., Cook, P. R. and Hoffman, M. (2010), Bayesian nonparametric matrix factorization for recorded music, *in* 'Proceedings of the 27th International Conference on Machine Learning', ICML '10, JMLR.org, Haifa, Israel, pp. 439–446.

Blei, D. M. and Frazier, P. I. (2011), 'Distance dependent chinese restaurant processes', *The Journal of Machine Learning Research* **12**, 2461–2488.

Blei, D. M., Griffiths, T. L. and Jordan, M. I. (2010), 'The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies', *Journal of the ACM* **57**(2), 7.

Blei, D. M., Jordan, M. I. et al. (2006), 'Variational inference for dirichlet process mixtures', *Bayesian Analysis* **1**(1), 121–143.

Blei, D. M., Ng, A. Y. and Jordan, M. I. (2003), 'Latent dirichlet allocation', *Journal of Machine Learning Research* **3**, 993–1022.

Blunsom, P. and Cohn, T. (2011), A hierarchical pitman-yor process hmm for unsupervised part of speech induction, *in* 'Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1', ACL '11, Association for Computational Linguistics, Association for Computational Linguistics, Portland, Oregon, USA, pp. 865–874.

Bouguila, N., Ziou, D. and Vaillancourt, J. (2004), 'Unsupervised learning of a finite mixture model based on the dirichlet distribution and its application', *IEEE Transactions on Image Processing* **13**(11), 1533–1543.

Broderick, T., Mackey, L., Paisley, J. and Jordan, M. (2015), 'Combinatorial clustering and the beta negative binomial process', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 290–306.

Bryant, M. and Sudderth, E. B. (2012), Truly nonparametric online variational inference for hierarchical dirichlet processes, *in* '26th Annual Conference on Neural Information Processing Systems', NIPS '12, JMLR.org, Lake Tahoe, Nevada, USA, pp. 2699–2707.

Campbell, T., Straub, J., III, J. W. F. and How, J. P. (2015), Streaming, distributed variational inference for bayesian nonparametrics, *in* 'Annual Conference on Neural Information Processing Systems', NIPS '15, JMLR.org, Montreal, Quebec, Canada, pp. 280–288.

Canini, K. R. and Griffiths, T. L. (2011), A nonparametric bayesian model of multi-level category learning, *in* 'Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence', AAAI '11, AAAI Press, San Francisco, California, pp. 307–312.

Caron, F., Davy, M. and Doucet, A. (2007), Generalized polya urn for time-varying dirichlet process mixtures, *in* 'Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence', UAI '07, AUAI Press, pp. 33–40.

Chang, J. and Blei, D. M. (2009), Relational topic models for document networks, *in* 'Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics', AISTATS '09, JMLR.org, Clearwater Beach, Florida, USA, pp. 81–88.

Chang, J., Blei, D. M. et al. (2010), 'Hierarchical relational models for document networks', *The Annals of Applied Statistics* **4**(1), 124–150.

Chang, J. and Fisher III, J. W. (2013), Parallel sampling of dp mixture models using sub-cluster splits, *in* '27th Annual Conference on Neural Information Processing Systems', NIPS '13, Lake Tahoe, Nevada, USA, pp. 620–628.

Chen, B., Polatkan, G., Sapiro, G., Carin, L. and Dunson, D. B. (2011), The hierarchical beta process for convolutional factor analysis and deep learning, *in* 'Proceedings of the 28th International Conference on Machine Learning', ICML '11, Omnipress, Bellevue, Washington, USA, pp. 361–368.

Chen, C., Buntine, W., Ding, N., Xie, L. and Du, L. (2015), 'Differential topic models', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 230–242.

Chen, C., Ding, N. and Buntine, W. L. (2012), Dependent hierarchical normalized random measures for dynamic topic modeling, *in* 'Proceedings of the

29th International Conference on Machine Learning', ICML '12, JMLR.org, Edinburgh, Scotland, UK.

Chen, C., Rao, V., Buntine, W. and Whye Teh, Y. (2013), Dependent normalized random measures, *in* 'Proceedings of the 30th International Conference on Machine Learning', ICML '13, JMLR.org, Atlanta, GA, USA, pp. 969–977.

Chen, N., Zhu, J., Xia, F. and Zhang, B. (2015), 'Discriminative relational topic models', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(5), 973–986.

Chuang, S.-L. and Chien, L.-F. (2004), A practical web-based approach to generating topic hierarchy for text segments, *in* 'Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management', CIKM '04, ACM, New York, NY, USA, pp. 127–136.

Chung, Y. and Dunson, D. B. (2009), 'The local dirichlet process', *Annals of the Institute of Statistical Mathematics* **63**(1), 59–80.

Cong, H. and Tong, L. H. (2008), 'Grouping of triz inventive principles to facilitate automatic patent classification', *Expert Systems with Applications* **34**(1), 788 – 795.

Crouch, C. J. (1988), A cluster-based approach to thesaurus construction, *in* 'Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval', SIGIR '88, ACM, New York, NY, USA, pp. 309–320.

da Silva, A. R. F. (2007), 'A dirichlet process mixture model for brain mri tissue classification', *Medical Image Analysis* **11**(2), 169–182.

Dai, A. and Storkey, A. (2015), 'The supervised hierarchical dirichlet process', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 243–255.

Damlen, P., Wakefield, J. and Walker, S. (1999), 'Gibbs sampling for bayesian non-conjugate and hierarchical models by using auxiliary variables', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **61**(2), 331–344.

DeNero, J., Bouchard-Côté, A. and Klein, D. (2008), Sampling alignment structure under a bayesian translation model, *in* 'Proceedings of the Conference on Empirical Methods in Natural Language Processing', EMNLP '08, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 314–323.

Dhillon, I. S. (2001), Co-clustering documents and words using bipartite spectral graph partitioning, *in* 'Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '01, ACM, New York, NY, USA, pp. 269–274.

Ding, C. H. Q., Li, T. and Jordan, M. I. (2010), 'Convex and semi-nonnegative matrix factorizations', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(1), 45–55.

Ding, N., Xiang, R., Molloy, I., Li, N. et al. (2010), Nonparametric bayesian matrix factorization by power-ep, *in* 'Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics', AISTATS '10, JMLR.org, pp. 169–176.

Doshi, F., Miller, K., Gael, J. V. and Teh, Y. W. (2009), Variational inference for the indian buffet process, *in* 'Proceedings of the Twelfth International

Conference on Artificial Intelligence and Statistics', AISTATS '09, JMLR.org, Clearwater Beach, Florida, USA, pp. 137–144.

Doshi-Velez, F., Pfau, D., Wood, F. and Roy, N. (2015), 'Bayesian nonparametric methods for partially-observable reinforcement learning', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 394–407.

Dubey, K., Williamson, S. and Xing, E. P. (2014), Parallel markov chain monte carlo for pitman-yor mixture models, *in* 'Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence', UAI '14, AUAI Press, Quebec City, Quebec, Canada, pp. 142–151.

Dunson, D. B. (2010), *Bayesian Nonparametrics*, Cambridge University Press, chapter Nonparametric Bayes applications to biostatistics, pp. 223–273.

Dunson, D. B. and Park, J.-H. (2008), 'Kernel stick-breaking processes', *Biometrika* **95**(2), 307–323.

Fan, X., Cao, L., Xu, D. and Yi, R. (2015), 'Dynamic infinite mixed-membership stochastic blockmodel', *IEEE Transactions on Neural Networks and Learning Systems* **26**(9), 2072–2085.

Favaro, S., Lijoi, A. and Prünster, I. (2012), 'A new estimator of the discovery probability', *Biometrics* **68**(4), 1188–1196.

Fearnhead, P. (2004), 'Particle filters for mixture models with an unknown number of components', *Statistics and Computing* **14**(1), 11–21.

Ferguson, T. S. (1973), 'A bayesian analysis of some nonparametric problems', *The Annals of Statistics* **1**(2), 209–230.

Fersini, E., Messina, E. and Archetti, F. (2008), Granular modeling of web documents: Impact on information retrieval systems, *in* 'Proceedings of the 10th

ACM Workshop on Web Information and Data Management', WIDM '08, ACM, New York, NY, USA, pp. 111–118.

Foti, N. J., Futoma, J. D., Rockmore, D. N. and Williamson, S. (2013), A unifying representation for a class of dependent random measures, *in* 'Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics', AISTATS '13, JMLR.org, Scottsdale, AZ, USA, pp. 20–28.

Foti, N. and Williamson, S. (2015), 'A survey of non-exchangeable priors for bayesian nonparametric models', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 359–371.

Fox, E. B. (2009), Bayesian nonparametric learning of complex dynamical phenomena, PhD thesis, Massachusetts Institute of Technology.

Fox, E. B., Hughes, M. C., Sudderth, E. B., Jordan, M. I. et al. (2014), 'Joint modeling of multiple time series via the beta process with application to motion capture segmentation', *The Annals of Applied Statistics* **8**(3), 1281–1313.

Fox, E., Sudderth, E., Jordan, M. and Willsky, A. (2011*a*), 'Bayesian nonparametric inference of wwitching dynamic linear models', *IEEE Transactions on Signal Processing* **59**(4), 1569–1585.

Fox, E., Sudderth, E., Jordan, M. and Willsky, A. (2011*b*), 'A sticky hdp-hmm with application to speaker diarization', *Annals of Applied Statistics* **5**(2A), 1020–1056.

Gael, J. V., Teh, Y. W. and Ghahramani, Z. (2008), The infinite factorial hidden markov model, *in* 'Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems', NIPS '08, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 1697–1704.

Gael, J. V., Vlachos, A. and Ghahramani, Z. (2009), The infinite hmm for unsupervised pos tagging, *in* 'Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing', EMNLP '09, Association for Computational Linguistics, Singapore, pp. 678–687.

Gao, Z., Song, Y., Liu, S., Wang, H., Wei, H., Chen, Y. and Cui, W. (2011), Tracking and connecting topics via incremental hierarchical dirichlet processes, *in* 'The 11th IEEE International Conference on Data Mining', ICDM '11, IEEE, Vancouver, Canada, pp. 1056–1061.

Ge, H., Chen, Y., Wan, M. and Ghahramani, Z. (2015), Distributed inference for dirichlet process mixture models, *in* 'Proceedings of the 32nd International Conference on Machine Learning', ICML '15, JMLR.org, Lille, France, pp. 2276–2284.

Gelfand, A. E., Kottas, A. and MacEachern, S. N. (2005), 'Bayesian nonparametric spatial modeling with dirichlet process mixing', *Journal of the American Statistical Association* **100**(471), 1021–1035.

Gershman, S. J. and Blei, D. M. (2012), 'A tutorial on bayesian nonparametric models', *Journal of Mathematical Psychology* **56**(1), 1–12.

Gershman, S. J., Frazier, P. I. and Blei, D. M. (2015), 'Distance dependent infinite latent feature models', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 334–345.

Ghahramani, Z., Jordan, M. I. and Adams, R. P. (2010), Tree-structured stick breaking for hierarchical data, *in* '24th Annual Conference on Neural Information Processing Systems', NIPS '10, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 19–27.

Ghosal, S. (2010), *Bayesian Nonparametrics*, Vol. 2, Cambridge University Press, chapter The Dirichlet process, related priors and posterior asymptotics, p-p. 35–79.

Gibaja, E. and Ventura, S. (2015), 'A tutorial on multilabel learning', *ACM Computing Surveys* **47**(3), 52:1–52:38.

Goldwater, S., Griffiths, T. L. and Johnson, M. (2009), 'A bayesian framework for word segmentation: Exploring the effects of context', *Cognition* **112**(1), 21 – 54.

Gonzalez, J., Low, Y., Gretton, A. and Guestrin, C. (2011), Parallel gibbs sampling: From colored fields to thin junction trees, *in* 'Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics', AISTATS '11, JMLR.org, Fort Lauderdale, USA, pp. 324–332.

Goyal, P., Behera, L. and McGinnity, T. M. (2013), 'A context-based word indexing model for document summarization', *IEEE Transactions on Knowledge and Data Engineering* **25**(8), 1693–1705.

Griffin, J. E., Kolossiatis, M. and Steel, M. F. J. (2013), 'Comparing distributions by using dependent normalized random-measure mixtures', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **75**(3), 499–529.

Griffin, J. E. and Steel, M. J. (2006), 'Order-based dependent dirichlet processes', *Journal of the American Statistical Association* **101**(473), 179–194.

Griffiths, T. L. and Ghahramani, Z. (2005), Infinite latent feature models and the indian buffet process, *in* 'Neural Information Processing Systems', NIPS '05, MIT Press, Vancouver, British Columbia, Canada, pp. 475–482.

Griffiths, T. L. and Ghahramani, Z. (2011), 'The indian buffet process: An introduction and review', *The Journal of Machine Learning Research* **12**, 1185–1224.

Griffiths, T. L. and Steyvers, M. (2004), 'Finding scientific topics', *Proceedings of the National Academy of Sciences* **101**(suppl 1), 5228–5235.

Guo, Z., Zhang, Z., Zhu, S., Chi, Y. and Gong, Y. (2014), 'A two-level topic model towards knowledge discovery from citation networks', *IEEE Transactions on Knowledge and Data Engineering* **26**(4), 780–794.

Gupta, S. K., Phung, D. Q. and Venkatesh, S. (2012), A slice sampler for restricted hierarchical beta process with applications to shared subspace learning, *in* 'Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence', UAI '12, AUAI Press, Catalina Island, CA, USA, pp. 316–325.

Gupta, V. and Lehal, G. S. (2009), 'A survey of text mining techniques and applications', *Journal of Emerging Technologies in Web Intelligence* **1**(1), 60–76.

Haines, T. S. F. and Xiang, T. (2014), 'Background subtraction with dirichlet process mixture models', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **36**(4), 670–683.

Han, J., Kamber, M. and Pei, J. (2011), *Data Mining: Concepts and Techniques*, Elsevier.

Hannah, L. A., Blei, D. M. and Powell, W. B. (2011), 'Dirichlet process mixtures of generalized linear models', *Journal of Machine Learning Research* **12**, 1923–1953.

He, L., Qi, H. and Zaretzki, R. (2013), Beta process joint dictionary learning for coupled feature spaces with application to single image super-resolution, *in* 'Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition', CVPR '13, IEEE, Portland, OR, USA, pp. 345–352.

He, Z., Xie, S., Zdunek, R., Zhou, G. and Cichocki, A. (2011), 'Symmetric nonnegative matrix factorization: Algorithms and applications to probabilistic clustering', *IEEE Transactions on Neural Networks* **22**(12), 2117–2131.

Heaukulani, C., Knowles, D. A. and Ghahramani, Z. (2014), Beta diffusion trees, *in* 'Proceedings of the 31th International Conference on Machine Learning', ICML '14, JMLR.org, Beijing, China, pp. 1809–1817.

Heiler, M. and Schnörr, C. (2006), 'Learning sparse representations by nonnegative matrix factorization and sequential cone programming', *The Journal of Machine Learning Research* **7**, 1385–1407.

Hjort, N. L. (1990), 'Nonparametric bayes estimators based on beta processes in models for life history data', *The Annals of Statistics* **18**(3), 1259–1294.

Hoffman, M. D., Blei, D. M. and Cook, P. R. (2008), Content-based musical similarity computation using the hierarchical dirichlet process, *in* '9th International Conference on Music Information Retrieval', ISMIR '08, Drexel University, Philadelphia, PA, USA, pp. 349–354.

Hoffman, M. D., Blei, D. M., Wang, C. and Paisley, J. (2013), 'Stochastic variational inference', *The Journal of Machine Learning Research* **14**(1), 1303–1347.

Hofmann, T. (1999), Probabilistic latent semantic indexing, *in* 'Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval', SIGIR '99, ACM, ACM, Berkeley, California, USA, pp. 50–57.

Huelsenbeck, J. P., Jain, S., Frost, S. W. and Pond, S. L. K. (2006), 'A dirichlet process model for detecting positive selection in protein-coding dna sequences', *Proceedings of the National Academy of Sciences* **103**(16), 6263–6268.

Iwata, T., Shah, A. and Ghahramani, Z. (2013), Discovering latent influence in online social activities via shared cascade poisson processes, *in* 'Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '13, ACM, Chicago, IL, USA, pp. 266–274.

Jbabdi, S., Woolrich, M. and Behrens, T. (2009), 'Multiple-subjects connectivity-based parcellation using hierarchical dirichlet process mixture models', *NeuroImage* **44**(2), 373 – 384.

Johnson, M. J. and Willsky, A. S. (2013), 'Bayesian nonparametric hidden semi-markov models', *Journal of Machine Learning Research* **14**(1), 673–701.

joon Kim, H. and goo Lee, S. (2003), 'Building topic hierarchy based on fuzzy relations', *Neurocomputing* **51**(0), 481 – 486.

Kalli, M., Griffin, J. E. and Walker, S. G. (2011), 'Slice sampling mixture models', *Statistics and Computing* **21**(1), 93–105.

Kim, H., Sun, Y., Hockenmaier, J. and Han, J. (2012), Etm: Entity topic models for mining documents associated with entities, *in* 'Proceedings of the 2012 IEEE 12th International Conference on Data Mining', ICDM '12, IEEE, Washington, DC, USA, pp. 349–358.

Kim, J. and Park, H. (2008), Sparse nonnegative matrix factorization for clustering, Technical report.

Kingman, J. (1982), 'The coalescent', *Stochastic Processes and their Applications* **13**(3), 235 – 248.

Kingman, J. F. C. (1967), 'Completely random measures.', *Pacific Journal of Mathematics* **21**(1), 59–78.

Kingman, J. F. C. (1992), *Poisson Processes*, Vol. 3, Oxford university press.

Klimt, B. and Yang, Y. (2004), *Machine Learning: ECML 2004: 15th European Conference on Machine Learning*, Springer Berlin Heidelberg, Berlin, Heidelberg, chapter The Enron Corpus: A New Dataset for Email Classification Research, pp. 217–226.

Knowles, D. A. and Ghahramani, Z. (2015), 'Pitman yor diffusion trees for bayesian hierarchical clustering', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 271–289.

Kurihara, K., Welling, M. and Teh, Y. W. (2007), Collapsed variational dirichlet process mixture models, *in* 'Proceedings of the 20th International Joint Conference on Artificial Intelligence', Vol. 7 of *IJCAI '07*, AAAI Press, Hyderabad, India, pp. 2796–2801.

Kurihara, K., Welling, M. and Vlassis, N. A. (2006), Accelerated variational dirichlet process mixtures, *in* 'Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems', NIPS '06, MIT Press, Vancouver, British Columbia, Canada, pp. 761–768.

Li, S. Z. (1995), *Markov Random Field Modeling in Computer Vision*, Springer-Verlag New York, Inc.

Li, T., Zhu, S. and Ogihara, M. (2003), Topic hierarchy generation via linear discriminant projection, *in* 'Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval', SIGIR '03, ACM, Toronto, Canada, pp. 421–422.

Li, W. and McCallum, A. (2006), Pachinko allocation: Dag-structured mixture models of topic correlations, *in* 'Proceedings of the 23rd International Conference on Machine Learning', ICML '06, ACM, Pittsburgh, Pennsylvania, USA, pp. 577–584.

Liang, D., Hoffman, M. D. and Ellis, D. P. (2013), Beta process sparse nonnegative matrix factorization for music., *in* 'Proceedings of the 14th International Society for Music Information Retrieval Conference', ISMIR '13, Curitiba, Brazil, pp. 375–380.

Lijoi, A., Mena, R. H. and Prünster, I. (2007), 'A bayesian nonparametric method for prediction in est analysis', *BMC Bioinformatics* **8**(1), 1–10.

Lin, C. and He, Y. (2009), Joint sentiment/topic model for sentiment analysis, *in* 'Proceedings of the 18th ACM Conference on Information and Knowledge Management', CIKM '09, ACM, ACM, Hong Kong, China, pp. 375–384.

Lin, D., Grimson, E. and III, J. W. F. (2010), Construction of dependent dirichlet processes based on poisson processes, *in* '24th Annual Conference on Neural Information Processing Systems', NIPS '10, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 1396–1404.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N. and Watkins, C. (2002), 'Text classification using string kernels', *The Journal of Machine Learning Research* **2**, 419–444.

Lovell, D., Adams, R. P. and Mansingka, V. (2012), Parallel markov chain monte carlo for dirichlet process mixtures, *in* 'Workshop on Big Learning, 26th Annual Conference on Neural Information Processing Systems', NIPS '12, Lake Tahoe, Nevada, USA.

Ma, Z., Rana, P. K., Taghia, J., Flierl, M. and Leijon, A. (2014), 'Bayesian estimation of dirichlet mixture model with variational inference', *Pattern Recognition* **47**(9), 3143 – 3157.

MacEachern, S. N. (1999), Dependent nonparametric processes, *in* 'ASA proceedings of the section on Bayesian statistical science', Alexandria, Virginia. Virginia: American Statistical Association; 1999, American Statistical Association, pp. 50–55.

MacEachern, S. N., Clyde, M. and Liu, J. S. (1999), 'Sequential importance sampling for nonparametric bayes models: The next generation', *Canadian Journal of Statistics* **27**(2), 251–267.

Maddison, C. J., Tarlow, D. and Minka, T. (2014), A* sampling, *in* 'Annual Conference on Neural Information Processing Systems', NIPS '14, JMLR.org, Montreal, Quebec, Canada, pp. 3086–3094.

Madjarov, G., Kocev, D., Gjorgjevikj, D. and DÅ¿eroski, S. (2012), 'An extensive experimental comparison of methods for multi-label learning', *Pattern Recognition* **45**(9), 3084 – 3104. Best Papers of Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'2011).

Mahmud, M. (2010), Constructing states for reinforcement learning, *in* 'Proceedings of the 27th International Conference on Machine Learning', ICML '10, Omnipress, Haifa, Israel, pp. 727–734.

McCullagh, P. (1984), 'Generalized linear models', *European Journal of Operational Research* **16**(3), 285–292.

Michael, T. and Jordan, I. (1994), Reinforcement learning algorithm for partially observable markov decision problems, *in* 'NIPS Conference', NIPS '94, MIT Press, Denver, CO, USA, pp. 345–352.

Mimno, D., Li, W. and McCallum, A. (2007), Mixtures of hierarchical topics with pachinko allocation, *in* 'Proceedings of the 24th International Conference on Machine Learning', ICML '07, ACM, Corvalis, Oregon, USA, pp. 633–640.

Miner, G. (2012), *Practical Text Mining and Statistical Analysis for Non-structured Text Data Applications*, Academic Press.

Minka, T. (2004), Power ep, Technical report, Technical report, Microsoft Research, Cambridge.

Nakano, M., Ohishi, Y., Kameoka, H., Mukai, R. and Kashino, K. (2012), Bayesian nonparametric music parser, *in* '2012 IEEE International Conference on Acoustics, Speech and Signal Processing', ICASSP '12, IEEE, Kyoto, Japan, pp. 461–464.

Neal, R. M. (2003*a*), 'Density modeling and clustering using dirichlet diffusion trees', *Bayesian Statistics* **7**, 619–629.

Neal, R. M. (2003*b*), 'Slice sampling', *Annals of Statistics* **31**(3), 705–767.

Neiswanger, W., Wang, C. and Xing, E. (2014), Embarrassingly parallel variational inference in nonconjugate models, *in* 'Workshop on Advanced Variational Inference', NIPS '14, JMLR.org, Montreal, Quebec, Canada, pp. 1–18.

Olkin, I. and Liu, R. (2003), 'A bivariate beta distribution', *Statistics & Probability Letters* **62**(4), 407–412.

Orbanz, P. and Teh, Y. W. (2010), *Encyclopedia of Machine Learning*, Springer US, Boston, MA, chapter Bayesian nonparametric models, pp. 81–89.

Paisley, J., Wang, C., Blei, D. and Jordan, M. (2015), 'Nested hierarchical dirichlet processes', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 256–270.

Park, H. W. (2003), 'Hyperlink network analysis: A new method for the study of social structure on the web', *Connections* **25**(1), 49–61.

Perotte, A. J., Wood, F., Elhadad, N. and Bartlett, N. (2011), Hierarchically supervised latent dirichlet allocation, *in* '25th Annual Conference on Neural Information Processing Systems', NIPS '11, JMLR.org, Granada, Spain, pp. 2609–2617.

Pestian, J. P., Brew, C., Matykiewicz, P., Hovermale, D., Johnson, N., Cohen, K. B. and Duch, W. (2007), A shared task involving multi-label classification of clinical free text, *in* 'Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing', BioNLP '07, Association for Computational Linguistics, Stroudsburg, PA, USA, pp. 97–104.

R. Daniel Mauldin, William D. Sudderth, S. C. W. (1992), 'Polya trees and random distributions', *The Annals of Statistics* **20**(3), 1203–1221.

Rabiner, L. R. (1989), 'A tutorial on hidden markov models and selected applications in speech recognition', *Proceedings of the IEEE* **77**(2), 257–286.

Ramage, D., Hall, D., Nallapati, R. and Manning, C. D. (2009), Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora, *in* 'Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1', EMNLP '09, Association for Computational Linguistics, Association for Computational Linguistics, Singapore, pp. 248–256.

Rao, V. and Teh, Y. W. (2009), Spatial normalized gamma processes, *in* '23rd Annual Conference on Neural Information Processing Systems', NIPS '09, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 1554–1562.

Rasmussen, C. E. (1999), The infinite gaussian mixture model, *in* 'Advances in Neural Information Processing Systems', NIPS '99, The MIT Press, Denver, CO, USA, pp. 554–560.

Reisinger, J. and Paşca, M. (2009), Latent variable models of concept-attribute attachment, *in* 'Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2', ACL/IJCNLP '09, Association for Computational Linguistics, Association for Computational Linguistics, Singapore, pp. 620–628.

Ren, L., Dunson, D. B. and Carin, L. (2008), The dynamic hierarchical dirichlet process, *in* 'Proceedings of the 25th International Conference on Machine Learning', ICML '08, ACM, Helsinki, Finland, pp. 824–831.

Ren, L., Dunson, D., Lindroth, S. and Carin, L. (2010), 'Dynamic nonparametric bayesian models for analysis of music', *Journal of the American Statistical Association* **105**(490), 458–472.

Ren, L., Wang, Y., Carin, L. and Dunson, D. B. (2011), The kernel beta process, *in* '25th Annual Conference on Neural Information Processing Systems', NIPS '11, Granada, Spain, pp. 963–971.

Rosen-Zvi, M., Chemudugunta, C., Griffiths, T., Smyth, P. and Steyvers, M. (2010), 'Learning author-topic models from text corpora', *ACM Transactions on Information Systems* **28**(1), 4:1–4:38.

Rosen-Zvi, M., Griffiths, T., Steyvers, M. and Smyth, P. (2004), The author-topic model for authors and documents, *in* 'Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence', UAI '04, AUAI Press, Arlington, Virginia, United States, pp. 487–494.

Rossi, R. G. and Rezende, S. O. (2011), Building a topic hierarchy using the bag-of-related-words representation, *in* 'Proceedings of the 11th ACM Symposium on Document Engineering', DocEng '11, ACM, Mountain View, CA, USA, pp. 195–204.

Roy, D. M. and Teh, Y. W. (2008), The mondrian process, *in* 'Proceedings of the Twenty-Second Annual Conference on Neural Information Processing Systems', NIPS '08, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 1377–1384.

Roychowdhury, A. and Kulis, B. (2015), Gamma processes, stick-breaking, and variational inference, *in* 'Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics', AISTATS '15, JMLR.org, San Diego, California, USA, pp. 800–808.

Sandler, R. and Lindenbaum, M. (2011), 'Nonnegative matrix factorization with earth mover's distance metric for image analysis', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(8), 1590–1602.

Sato, I., Kurihara, K. and Nakagawa, H. (2012), Practical collapsed variational bayes inference for hierarchical dirichlet process, *in* 'Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining', KDD '12, ACM, ACM, Beijing, China, pp. 105–113.

Seeger, M. (2004), 'Gaussian processes for machine learning', *International Journal of Neural Systems* **14**(02), 69–106.

Sen, P., Namata, G. M., Bilgic, M., Getoor, L., Gallagher, B. and Eliassi-Rad, T. (2008), 'Collective classification in network data', *AI Magazine* **29**(3), 93–106.

Sethuraman, J. (1994), 'A constructive definition of dirichlet priors', *Statistica Sinica* **4**, 639–650.

Shahbaba, B. and Neal, R. (2009), 'Nonlinear models using dirichlet process mixtures', *Journal of Machine Learning Research* **10**, 1829–1850.

Shan, H. and Banerjee, A. (2008), Bayesian co-clustering, *in* 'The Eighth IEEE International Conference on Data Mining', ICDM '08, IEEE, Pisa, Italy, pp. 530–539.

Slutsky, A., Hu, X. and An, Y. (2013), Tree labeled lda: A hierarchical model for web summaries, *in* 'IEEE International Conference on Big Data', BigData '13, IEEE, Santa Clara, CA, USA, pp. 134–140.

Smyth, P., Welling, M. and Asuncion, A. U. (2009), Asynchronous distributed learning of topic models, *in* '23rd Annual Conference on Neural Information Processing Systems', NIPS '09, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 81–88.

Steinhardt, J. and Ghahramani, Z. (2012), Flexible martingale priors for deep hierarchies, *in* 'Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics', AISTATS '12, JMLR.org, La Palma, Canary Islands, pp. 1108–1116.

Steyvers, M., Smyth, P., Rosen-Zvi, M. and Griffiths, T. (2004), Probabilistic author-topic models for information discovery, *in* 'Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '04, ACM, Seattle, WA, USA, pp. 306–315.

Sun, Y., Deng, H. and Han, J. (2012), *Mining Text Data*, Springer US, Boston, MA, chapter Probabilistic models for text mining, pp. 259–295.

Tamara Broderick, M. I. J. and Pitman, J. (2012), 'Beta processes, stick-breaking and power laws', *Bayesian Analysis* **7**(2), 439–476.

Tanner, M. A. and Wong, W. H. (2010), 'From em to data augmentation: The emergence of mcmc bayesian computation in the 1980s', *Statistical Science* **25**(4), 506–516.

Teh, Y., Daumé, H. and Roy, D. (2007), Bayesian agglomerative clustering with coalescents, *in* 'Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems', NIPS '07, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 1473–1480.

Teh, Y. W. (2006), A hierarchical bayesian language model based on pitman-yor processes, *in* 'Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics', ACL '06, Association for Computational Linguistics, Association for Computational Linguistics, Sydney, NSW, Australia, pp. 985–992.

Teh, Y. W. (2010), Dirichlet process, *in* C. Sammut and G. Webb, eds, 'Encyclopedia of Machine Learning', Springer US, pp. 280–287.

Teh, Y. W., Blundell, C. and Elliott, L. (2011), Modelling genetic variations using fragmentation-coagulation processes, *in* '25th Annual Conference on Neural Information Processing Systems', NIPS '11, JMLR.org, Granada, Spain, pp. 819–827.

Teh, Y. W., Görür, D. and Ghahramani, Z. (2007), Stick-breaking construction for the indian buffet process, *in* 'Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics', AISTATS '07, JMLR.org, San Juan, Puerto Rico, pp. 556–563.

Teh, Y. W., Kurihara, K. and Welling, M. (2007), Collapsed variational inference for hdp, *in* 'Proceedings of the Twenty-First Annual Conference on Neural Information Processing Systems', NIPS '07, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 1481–1488.

Temperley, D. (2007), *Music and Probability*, MIT Press.

Thibaux, R. and Jordan, M. I. (2007), Hierarchical beta processes and the indian buffet process, *in* 'Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics', AISTATS '07, JMLR.org, San Juan, Puerto Rico, pp. 564–571.

Trivedi, P. K. and Zimmer, D. M. (2007), *Copula Modeling: An Introduction for Practitioners*, Now Publishers Inc.

Van Dyk, D. A. and Meng, X.-L. (2001), 'The art of data augmentation', *Journal of Computational and Graphical Statistics* **10**(1), 1–50.

Wang, C. and Blei, D. M. (2009), Variational inference for the nested chinese restaurant process, *in* '23rd Annual Conference on Neural Information Processing Systems', NIPS '09, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 1990–1998.

Wang, C. and Blei, D. M. (2012), Truncation-free online variational inference for bayesian nonparametric models, *in* '26th Annual Conference on Neural Information Processing Systems', NIPS '12, Lake Tahoe, Nevada, USA, pp. 413–421.

Wang, C., Paisley, J. W. and Blei, D. M. (2011), Online variational inference for the hierarchical dirichlet process, *in* 'Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics', AISTATS '11, JMLR.org, Fort Lauderdale, USA, pp. 752–760.

Wang, X. and McCallum, A. (2006), Topics over time: A non-markov continuous-time model of topical trends, *in* 'Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '06, ACM, Philadelphia, PA, USA, pp. 424–433.

Wang, Y. and Carin, L. (2012), Levy measure decompositions for the beta and gamma processes, *in* 'Proceedings of the 29th International Conference on Machine Learning', ICML '12, ACM, Edinburgh, Scotland, UK.

William B. Frakes, R. B.-Y. (1992), *Information Retrieval: Data Structures and Algorithms*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Williamson, S., Dubey, A. and Xing, E. (2013), Parallel markov chain monte carlo for nonparametric mixture models, *in* 'Proceedings of the 30th International Conference on Machine Learning', ICML '13, JMLR.org, Atlanta, GA, USA, pp. 98–106.

Williamson, S., Orbanz, P. and Ghahramani, Z. (2010), Dependent indian buffet processes, *in* 'Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics', AISTATS '10, JMLR.org, Sardinia, Italy, pp. 924–931.

Willsky, A. S., Sudderth, E. B., Jordan, M. I. and Fox, E. B. (2009), Nonparametric bayesian learning of switching linear dynamical systems, *in* '23rd Annual Conference on Neural Information Processing Systems', NIPS '09, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 457–464.

Wood, F. and Griffiths, T. L. (2006), Particle filtering for nonparametric bayesian matrix factorization, *in* 'Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems', NIPS '06, MIT Press, Vancouver, British Columbia, Canada, pp. 1513–1520.

Wulsin, D. F., Fox, E. B. and Litt, B. (2014), 'Modeling the complex dynamics and changing correlations of epileptic events', *Artificial Intelligence* **216**(0), 55 – 75.

Xu, T., Zhang, Z., Yu, P. and Long, B. (2008), Evolutionary clustering by hierarchical dirichlet process with hidden markov state, *in* 'The Eighth IEEE International Conference on Data Mining', ICDM '08, IEEE, Pisa, Italy, pp. 658–667.

Xuan, J., Lu, J., Zhang, G. and Luo, X. (2015), 'Topic model for graph mining', *IEEE Transactions on Cybernetics* **45**(12), 2792–2803.

Xuan, J., Lu, J., Zhang, G., Xu, R. and Luo, X. (2015), Infinite author topic model based on mixed gamma-negative binomial process, *in* 'The 15th IEEE International Conference on Data Mining', ICDM '15, IEEE, Atlantic City, New Jersey, USA, pp. 489–498.

Yang, Z., Zhou, G., Xie, S., Ding, S., Yang, J.-M. and Zhang, J. (2011), 'Blind spectral unmixing based on sparse nonnegative matrix factorization', *IEEE Transactions on Image Processing* **20**(4), 1112–1125.

Ye, M., Qian, Y. and Zhou, J. (2015), 'Multitask sparse nonnegative matrix factorization for joint spectral-spatial hyperspectral imagery denoising', *IEEE Transactions on Geoscience and Remote Sensing* **53**(5), 2621–2639.

Yerebakan, H. Z., Rajwa, B. and Dundar, M. (2014), The infinite mixture of infinite gaussian mixtures, *in* 'Annual Conference on Neural Information Processing Systems', NIPS '14, Montreal, Quebec, Canada, pp. 28–36.

Yu, S.-Z. (2010), 'Hidden semi-markov models', *Artificial Intelligence* **174**(2), 215 – 243. Special Review Issue.

Zhang, C., Ek, C. H., Gratal, X., Pokorny, F. T. and Kjellstrom, H. (2013), Supervised hierarchical dirichlet processes with variational inference, *in* 'IEEE International Conference on Computer Vision Workshops', ICCVW '13, IEEE, Sydney, NSW, pp. 254–261.

Zhang, J., Song, Y., Zhang, C. and Liu, S. (2010), Evolutionary hierarchical dirichlet processes for multiple correlated time-varying corpora, *in* 'Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '10, ACM, Washington, DC, USA, pp. 1079–1088.

Zhang, M. (2011), Lift: Multi-label learning with label-specific features, *in* 'Proceedings of the 22nd International Joint Conference on Artificial Intelligence', IJCAI '11, AAAI Press, arcelona, Catalonia, Spain, pp. 1609–1614.

Zhang, M.-L. and Wu, L. (2015), 'Lift: Multi-label learning with label-specific features', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(1), 107–120.

Zhang, M.-L. and Zhang, K. (2010), Multi-label learning by exploiting label dependency, *in* 'Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining', KDD '10, ACM, Washington, DC, USA, pp. 999–1008.

Zhang, M.-L. and Zhou, Z.-H. (2014), 'A review on multi-label learning algorithms', *IEEE Transactions on Knowledge and Data Engineering* **26**(8), 1819–1837.

Zheng, J., Liu, S. and Ni, L. (2014), Effective mobile context pattern discovery via adapted hierarchical dirichlet processes, *in* 'IEEE 15th International Conference on Mobile Data Management', Vol. 1 of *MDM '14*, IEEE, Brisbane, Australia, pp. 146–155.

Zhou, M. and Carin, L. (2012), Augment-and-conquer negative binomial processes, *in* '26th Annual Conference on Neural Information Processing Systems', NIPS '12, Lake Tahoe, Nevada, USA, pp. 2546–2554.

Zhou, M. and Carin, L. (2015), 'Negative binomial process count and mixture modeling', *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(2), 307–320.

Zhou, M., Chen, H., Ren, L., Sapiro, G., Carin, L. and Paisley, J. W. (2009), Non-parametric bayesian dictionary learning for sparse image representations, *in* '23rd Annual Conference on Neural Information Processing Systems', NIPS '09, Curran Associates, Inc., Vancouver, British Columbia, Canada, pp. 2295–2303.

Zhou, M., Hannah, L., Dunson, D. B. and Carin, L. (2012), Beta-negative binomial process and poisson gactor snalysis, *in* 'Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics', AISTATS '12, JMLR.org, La Palma, Canary Islands, pp. 1462–1471.

Zhou, M., Yang, H., Sapiro, G., Dunson, D. B. and Carin, L. (2011), Dependent hierarchical beta process for image interpolation and denoising, *in* 'Proceedings of the 22nd International Joint Conference on Artificial Intelligence', IJCAI '11, Barcelona, Catalonia, Spain, pp. 883–891.

Zhu, J., Ahmed, A. and Xing, E. P. (2012), 'Medlda: Maximum margin supervised topic models', *Journal of Machine Learning Research* **13**(1), 2237–2278.

Zhu, X., Ming, Z.-Y., Zhu, X. and Chua, T.-S. (2013), Topic hierarchy construction for the organization of multi-source user generated contents, *in* 'Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval', SIGIR '13, ACM, Dubin, Ireland, pp. 233–242.

ABBREVIATIONS

ATM	Author topic model
BNL	Bayesian nonparametric learning
BNM	Bayesian nonparametric model
CHDP	Cooperative hierarchial Dirichlet process
CRF	Chinese restaurant franchise
CRM	Completely random measure
CRP	Chinese restaurant process
DMPM	Dirichlet mixture probability measure
DP	Dirichlet process
DTM	Deep topic model
dIBP	Dependent Indian buffet processes
FGM	Farlie-Gumbel-Morgenstern
GaP	Gamma process
GEM	Griffiths-Engen-McCloskey random measure
GP	Gaussian process
HDP	Hierarchial Dirichlet process
IBP	Indian buffet process
LDA	Latent Dirichlet allocation
IRP	International restaurant process
MCMC	Markov chain Monte Carlo

MGNBP	Mixed Gamma negative binomial process
MRF	Markov random field
M-H	Metropolis-Hastings sampling
NBP	Negative binomial process
NMF	Nonnegative matrix factorisation
NTR	Nonparametric relational topic model
RTM	Relational topic model
SMRF	Subsampling Markov random field