Structured Bayesian Latent Factor Models with Meta-data

He Zhao

BEng, MEng

A thesis submitted for the degree of Doctor of Philosophy at

Monash University in 2019

Faculty of Information Technology

Dedicated to my wife and my parents

Copyright Notice

Abstract

In the era of big data, huge amounts of data are being generated from the internet, social networks, phone apps, and so on, which creates high demand for powerful and efficient data analysis techniques. In areas such as collaborative filtering, text analysis, graph analysis, and bioinformatics, a large proportion of such data can be formulated into discrete matrices. For example, in recommender systems, users' shopping history can be represented as a count/binary item-user matrix, with each entry indicating whether or not a user has bought an item (or his/her purchase count); In text analysis, a collection of documents can be represented as a word-document count matrix with the bag-of-words assumption; In graph analysis, the interactions between the users in a social network be modelled by an adjacency matrix, each entry of which, for instance, captures whether a user follows another or how many times a user replies another's tweets.

Bayesian latent factor models have enjoyed great success in analysing the above kinds of discrete data, which are probabilistic generative models factorising the parameters of the distribution that generates data samples with low-dimensional stochastic latent representations. It is known that Bayesian latent factor models have appealing advantages on modelling high-dimensionality, data sparsity and missing data, which are common challenges in analysing internet generated data. In this thesis, I will elaborate on the details of the Bayesian latent factor models for multiple applications including text analysis, graph analysis, and multi-label problems, proposed in my PhD research. The novelties of the proposed approaches particularly focus on the following directions:

- Incorporating meta-data to help data analysis in the case where a large proportion of data are unobserved, such as leveraging node attributes to predict missing links and discovering latent communities in graph analysis.

- Discovering hierarchically structured representations of data, such as learning interpretable correlation structures of topics from text collections;

- Developing efficient inference algorithms that leverage the sparsity of data and meta-data.

In comparison to many state-of-the-art methods in the above areas, the proposed approaches have achieved not only better modelling performance and efficiency, but also preferable interpretability for intuitively understanding those data, which is an increasingly important property in machine learning and data mining.

Publications During Enrolment

- Zhao, H., Du, L., & Buntine, W. (2017, August). Leveraging node attributes for incomplete relational data. In *International Conference on Machine Learning* (pp. 4072-4081).
- Zhao, H., Du, L., Buntine, W., & Liu, G. (2017, November). MetaLDA: A topic model that efficiently incorporates meta information. In *IEEE International Conference on Data Mining* (pp. 635-644).
- Zhao, H., Du, L., & Buntine, W. (2017, November). A word embeddings informed focused topic model. In *Asian Conference on Machine Learning* (pp. 423-438).
- Zhao, H., Rai, P., Du, L., & Buntine, W. (2018, March). Bayesian multi-label learning with sparse features and labels, and label co-occurrences. In *International Conference on Artificial Intelligence and Statistics* (pp. 1943-1951).
- Zhao, H., Du, L., Buntine, W., & Liu, G. (2018). Leveraging external information in topic modelling. *Knowledge and Information Systems*, 1-33.
- Zhao, H., Du, L., Buntine, W., & Zhou, M. (2018, July). Inter and intra topic structure learning with word embeddings. In *International Conference on Machine Learning* (pp. 5887-5896).
- Zhao, H., Du, L., Buntine, W., & Zhou, M. (2018, December). Dirichlet belief networks for topic structure learning. In *Advances in Neural Information Processing Systems* (pp. 7966-7977).

Thesis including published works declaration

I hereby declare that this thesis contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

This thesis includes 7 original papers published in peer reviewed conferences and journals. The core theme of the thesis is structured Bayesian latent factor models with meta-data for text analysis, graph analysis, and multi-label learning problems. The ideas, development and writing up of all the papers in the thesis were the principal responsibility of myself, the student, working within the Faculty of Information Technology under the supervision of Dr Lan Du and Prof Wray Buntine.

The inclusion of co-authors reflects the fact that the work came from active collaboration between researchers and acknowledges input into team-based research.

In the case of Chapter 4, 5, 6, and 7, my contribution to the work involved the following:

Thesis Chapter	**Publication Title**	**Status** *(published, in press, accepted or returned for revision)*	**Nature and % of student contribution**	**Co-author name(s) Nature and % of Co-author's contribution***	**Co-author(s), Monash student Y/N***
4	Leveraging Node Attributes for Incomplete Relational Data	Accepted	60%. Proposing research ideas, implementing models, conducting experiments, writing papers	1) Lan Du, discussing idea, writing paper, 20% 2) Wray Buntine, discussing idea, writing paper, 20%	N
5	MetaLDA: A Topic Model that Efficiently Incorporates Meta information	Accepted	58%. Proposing research ideas, implementing models, conducting experiments, writing papers	1) Lan Du, discussing idea, writing paper, 20% 2) Wray Buntine, discussing idea, writing paper, 20% 3) Gang Liu, data preparation, %2	N
5	Leveraging External Information in Topic Modelling	Accepted	58%. Proposing research ideas, implementing models, conducting experiments, writing papers	1) Lan Du, discussing idea, writing paper, 20% 2) Wray Buntine, discussing idea, writing paper, 20% 3) Gang Liu, data preparation, %2	N
5	A Word Embeddings Informed Focused Topic Model	Accepted	60%. Proposing research ideas, implementing models, conducting experiments, writing papers	1) Lan Du, discussing idea, writing paper, 20% 2) Wray Buntine, discussing idea, writing paper, 20%	N
6	Inter and Intra Topic Structure Learning with Word Embeddings	Accepted	55%. Proposing research ideas, implementing models, conducting	1) Lan Du, discussing idea, writing paper, 15%	N

			experiments, writing papers	2) Wray Buntine, discussing idea, writing paper, 15% 3) Mingyuan Zhou, discussing idea, writing paper, 15%	
6	Dirichlet Belief Networks for Topic Structure Learning	Accepted	55%. Proposing research ideas, implementing models, conducting experiments, writing papers	1) Lan Du, discussing idea, writing paper, 15% 2) Wray Buntine, discussing idea, writing paper, 15% 3) Mingyuan Zhou, discussing idea, writing paper, 15%	N
7	Bayesian Multi-label Learning with Sparse Features and Labels, and Label Co-occurrences	Accepted	50%. Proposing research ideas, implementing models, conducting experiments, writing papers	1) Piyush Rai, discussing idea, conducting experiments, writing paper, 20% 2) Lan Du, discussing idea, writing paper, 15% 3) Wray Buntine, discussing idea, writing paper, 15%	N

I have / have not renumbered sections of submitted or published papers in order to generate a consistent presentation within the thesis.

Student signature: **Date:**

The undersigned hereby certify that the above declaration correctly reflects the nature and extent of the student's and co-authors' contributions to this work. In instances where I am not the responsible author I have consulted with the responsible author to agree on the respective contributions of the authors.

Main Supervisor signature: **Date:**

Acknowledgement

On the completion of this thesis, I would like to thank the people who have offered their kind help during my PhD study and in my life.

First and foremost, I give my most sincere gratitude to my PhD supervisors, Prof Wray Buntine and Dr Lan Du. I want to thank Wray for his acceptance to me as his PhD student, when I was experiencing a very hard time of my life and about to give up my dream of being a researcher. It was a life-changing opportunity for me. During my study, I have benefited so much from Wray's profound and deep knowledge, rigorous attitude, great passion and enthusiasm for research, which will continue inspiring me in my entire research career. As my supervisor, Wray always listens to me, thinks for me, respects my opinions, and gives me invaluable advice on research and life. I give my utmost thanks to Wray, for teaching, supervising, supporting me, and even working with me overnight to catch up with paper deadlines. I would also like to express my greatest gratitude to Lan for his kind supervision, support, and help. At the beginning of my PhD study, I had a little background in my research area. At that time, Lan patiently and constructively transferred his rich experience and in-depth knowledge to me without reservation, which opened the door to the world of statistical machine learning for me. I feel very grateful to him for giving me the privilege of asking questions and brainstorming research ideas at any time, making his office the place where many interesting ideas were generated. Moreover, I thank Lan for emphasising the importance of writing skills to me and consistently helping me improve mine, which I will be benefited from in my research career. Furthermore, I also gratefully appreciate Lan for being strict and rigorous to my research but being kind as a friend in my non-research life. Upon the completion of my PhD study, I cannot thank my perfect supervising team, Wray and Lan, enough for kindly being so supportive but allowing me to explore what I want and collaborate with others. In addition to supervisors, they are like friends of mine. It is definitely my pleasure and honour for having the opportunities of working with them and learning from them. What a wonderful and unforgettable journey!

Many thanks go to my collaborators, Prof Mingyuan Zhou at UT Austin and Prof Piyush Rai at IIT Kanpur, who I feel very fortunate to work with. Before I met Mingyuan and Piyush, I had been inspired by their brilliant works and my first paper was an extension to one of Mingyuan's inspiring methods. After having several nice chats with Mingyuan and Piyush in ICML conference, I was very lucky and honoured to start collaborating extensively with them since then. I have been greatly impressed and inspired by their deep knowledge, smartness, creativeness, kindness, and hard-working. The collaborations with them make a great contribution to this thesis research. Specifically, I would like to appreciate Piyush's invitation to me for jointly working on our AISTATS paper and his invaluable guidance and help thereafter. I want to give my big thanks to Mingyuan for his kind host of my short visit to UT Austin, where we brainstormed many promising research ideas leading to several interesting papers later. Mingyuan is an incredibly smart and hard-working researcher with both deep understandings of theory and sharp insights of applications. He is a research role model of mine. I also want to express my great thanks to Mingyuan for his strong help and support for my job hunting.

I would like to thank my thesis panel members at Monash University, Prof Reza Haffari and Prof Dinh Phung, who are amazing researchers and supervisors, for their kind support and precious comments since the start of my PhD study. I also thank Reza for allowing me to join his reading group, his great help to my job hunting, and his trust on me for helping teach his course. For Dinh, I feel very grateful for his appreciation, the invaluable suggestions to my research career, and the invitation of my talk in his group.

I also own my thanks to Prof Changyou Chen at the University at Buffalo, the State University of New York, who hosted my short visit, where I learned a lot from him and his amazing research. I would like to appreciate Dr Hang Li at Bytedance AI Lab, for kindly hosting my short visit to his group, where I had a great time meeting the brilliant researchers and engineers. I also would like to thank Prof Anton van den Hengel and Prof Chunhua Shen at the University of Adelaide, for their invitation and host of my talk.

I also feel lucky for having Prof Sidan Du and Prof Ziqiang Wang at Nanjing University as the supervisors of my Master's study. I would like to thank them for their advising and kind support to my PhD applications, without which, I would not have had the opportunity to pursue my PhD.

I give my gratitude to my friends Ye Zhu and Dawei Sun for doing sports, playing video games, and hanging out with me. I thank Joan Capdevila Pujol for kindly treating and showing me around in Barcelona. We also had a great time working together. I also thank the people in the Faculty of IT, including but not limited to François Petitjean, Daniel Schmidt, Minyi Li, Ming Liu, and He Zhang.

At last, words cannot express my thanks to my family for their unconditioned love and support. I feel so grateful to my parents, who not only brought me to the world and raised me but also understand me as my closest friends. My most special thanks go to my wife, Wenyue, for her incomparable love and going through thick and thin with me in more than a decade. She completes me.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the era of big data, huge amounts of data are being generated from the internet, social networks, phone apps, and so on. These data create high demand for powerful and efficient data analysis techniques. This thesis particularly focuses on large-scale, high-dimensional, sparse, discrete data, which are ubiquitous in many applications, such as collaborative filtering, text analysis, graph analysis, and bioinformatics. For example:

- In recommender systems, users' shopping history can be represented as a count/binary item-user matrix, with each entry indicating whether or not a user has bought an item (or his/her purchase count). Analysing such data is important for discovering user preferences and item properties, and making recommendations.

- In text analysis, a collection of documents can be represented as a word-document count matrix with the bag-of-words assumption. Modelling such bag-of-words data can help us understand the semantic content of documents.

- In graph analysis, the interactions between the users in a social network be modelled by an adjacency matrix, each entry of which, for instance, captures whether a user follows another or how many times a user replies another's tweets. Modelling those data is essential to tasks such as discovering communities and studying how information is propagated in social networks.

To analyse the above kinds of data, considerable research efforts have been devoted to proposing machine learning and data mining tools, among which, *Bayesian probabilistic models* have been one of the most important families. A Bayesian model usually learns the distribution that generates data samples with latent variables, which can be used to not only analyse data properties but also generate new data samples. My research in this thesis focuses on a particular series of Bayesian probabilistic models for modelling data formatted in matrices, i.e., *Bayesian Latent Factor Models* (BLFMs) [Blei et al., 2003, Canny, 2004, Mnih and Salakhutdinov, 2008, Zhou et al., 2012a], which factorise the parameters of the data distribution with low-dimensional stochastic latent representations. The basic framework of a BLFM can be demonstrated in Figure 1.1. BLFMs are Bayesian extensions of matrix factorisation, which is a primary approach for data analysis. It is known that BLFMs have appealing advantages on modelling high-dimensionality, data sparsity and missing data, which are common challenges in analysing internet generated data.

In this thesis research, I will present several new developments of BLFMs for discrete data, such as texts and graphs, focusing on improving existing methods with three general directions and motivations:

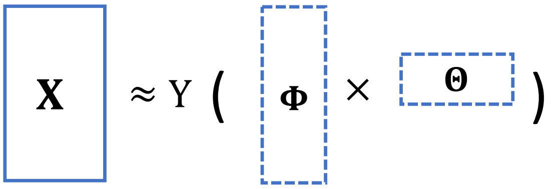
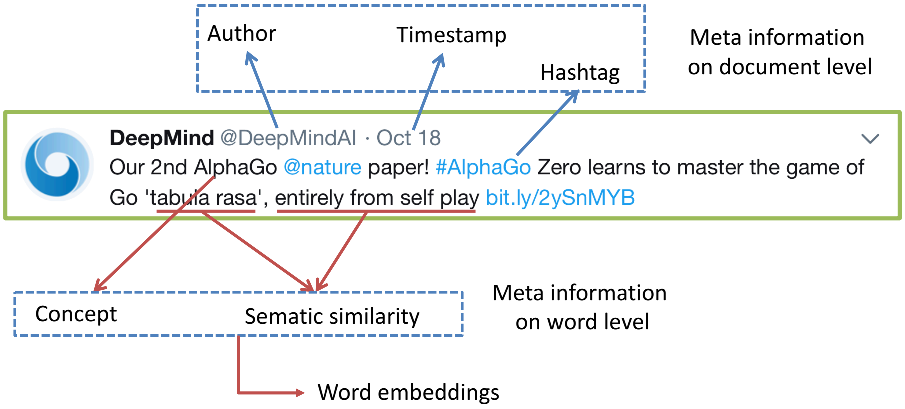


Figure 1.1: A basic demonstration of a BLFM. $\mathbf{X}$ *is the data matrix.* Y *is the distribution that generates* $\mathbf{X}$*, which is parametrised by a product of* $\boldsymbol{\Theta}$ *and* $\boldsymbol{\Phi}$*, the latent representations of* $\mathbf{X}$*.*



DeepMind @DeepMindAI · Oct 18
Our 2nd AlphaGo @nature paper! #AlphaGo Zero learns to master the game of Go 'tabula rasa', entirely from self play bit.ly/2ySnMYB

Figure 1.2: A demonstration of the meta-data of a tweet [Zhao et al., 2018a].

Effectively Leveraging meta-data Given a target data matrix that we would like to analyse, rich *meta-data* are usually accessible, which are able to help us obtain improved modelling performance and get better intuitive understandings of the target data matrix. Meta-data can serve as critical complementary information especially when the target data matrix is sparse or contains a large proportion of missing entries. For example, suppose we would like to understand the content of a tweet shown in Figure 1.2. Given that the length of the tweet is quite short (i.e., the data vector is very sparse), conventional BLFMs for text analysis may fail to capture its semantic meanings. However, if a model can leverage the rich meta-data associated with the tweet such as author, timestamp, word meanings, it would usually provide better performance. Therefore, one of the major directions of this research is developing effective approaches for BLFMs to incorporate various kinds of meta-data for different tasks on analysing discrete data.

Learning hierarchically structured latent representations with interpretability On the other hand, given the complexity of the target data matrix, it usually requires complicated data distributions to model it, which can be implemented by imposing hierarchically structured latent variables. hierarchically structured latent variables are required to not only improve modelling accuracy but also obtain better interpretability, which is a preferable property in many applications such as text analysis. For example, it is known that one important application of BLFMs for text analysis is discovering latent topics, which can be interpreted as semantic groups of words. Conventional BLFM approaches assume topics are independent and learn topics with flat structures of latent variables, which are not able to capture the structured correlations among topics, such as the example shown in Figure 1.3. Therefore, another major direction of this thesis research is learning interpretable hierarchically structured latent variables to obtain better modelling performance and better intuitive understandings for text data.

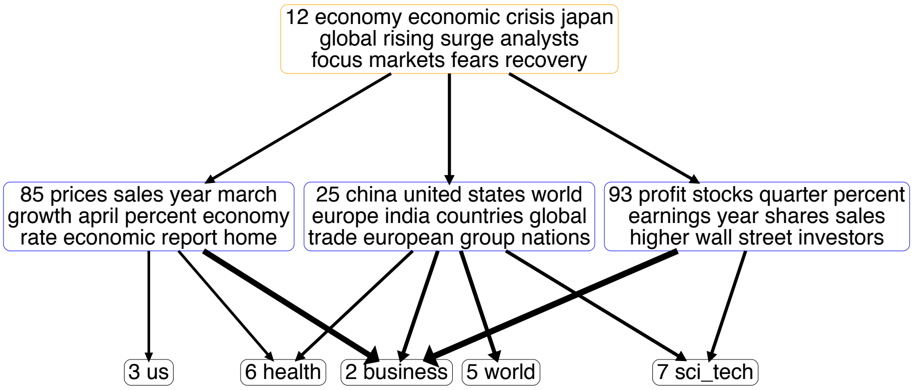

12 economy economic crisis japan global rising surge analysts focus markets fears recovery
85 prices sales year march growth april percent economy rate economic report home
25 china united states world europe india countries global trade european group nations
93 profit stocks quarter percent earnings year shares sales higher wall street investors
3 us
6 health
2 business
5 world
7 sci_tech

Figure 1.3: A demonstration of a topic hierarchy on a news article dataset [Zhao et al., 2018b]. The top orange topic is a general topic about economy connected by three more specific related topics covering different aspects of economy. The bottom are labels of the topics. Thicker arrows indicate stronger correlations.

Developing efficient learning algorithms large-scale data Usually, leveraging meta-data and learning interpretable hierarchically structured latent representations increase model complexity of BLFMs, which makes the learning of such models non-trivial especially on big data and prevents applications of those models. Fortunately, internet generated data such as texts and graphs can usually be formulated into matrices in large dimensions yet sparse. Therefore, the final direction of this research is developing efficient Bayesian inference algorithms for BLFMs by leveraging the sparsity in target data and meta-data.

Given the above three major motivations, the research theme of this thesis can be briefly summarised as developing BLFMs for discrete data, which are able to: **1)** incorporate the meta-data [Zhao et al., 2017a,c,b, 2018a,d] to help analyse the target data matrix, such as predicting missing links and discovering latent communities in graph analysis by leveraging node attributes; **2)** discover hierarchically structured representations [Zhao et al., 2018c,b], such as learning interpretable topic structures of documents; **3)** facilitate efficient inference algorithms [Zhao et al., 2017a,c, 2018d], such as leveraging data sparsity to speed up the training phrase. On analysing the above kinds of discrete data, the proposed approaches have achieved not only better modelling performance and efficiency, but also preferable interpretability, in the areas of text analysis [Zhao et al., 2017c,b, 2018a,c,b], graph analysis [Zhao et al., 2017a], and multi-label learning [Zhao et al., 2018d].

1.1 List of Thesis Publications

This thesis includes seven papers accepted for publication in peer-reviewed conference proceedings or journals. I am the first author and the principle innovator for all papers, which are listed in reverse chronological order:

1. **H. Zhao**, L. Du, W. Buntine, M. Zhou, "Dirichlet Belief Networks for Topic Structure Learning", in *Neural Information Processing Systems* (**NeurIPS**) 2018.

2. **H. Zhao**, L. Du, W. Buntine, M. Zhou, "Inter and Intra Topic Structure Learning with Word Embeddings", in *International Conference on Machine Learning* (**ICML**) 2018.

3. **H. Zhao**, L. Du, W. Buntine, G. Liu, "Leveraging External Information In Topic Modelling", in *Knowledge and Information Systems* (**KAIS**) 2018.

4. **H. Zhao**, P. Rai, L. Du, W. Buntine, "Bayesian Multi-label Learning with Sparse Features and Labels, and Label Co-occurrences", in *Artificial Intelligence and Statistics* (**AISTATS**) 2018.

5. **H. Zhao**, L. Du, W. Buntine, "A Word Embeddings Informed Focused Topic Model", in *Asian Conference on Machine Learning* (**ACML**) 2017.

6. **H. Zhao**, L. Du, W. Buntine, G. Liu, "MetaLDA: A Topic Model that Efficiently Incorporates Meta information", long paper in *International Conference on Data Mining* (**ICDM**) 2017.

7. **H. Zhao**, L. Du, W. Buntine, "Leveraging Node Attributes for Incomplete Relational Data", *International Conference on Machine Learning* (**ICML**) 2017.

1.2 Summary of Contributions

According to the research areas, the contributions of this thesis are categorised as follows:

Bayesian random graph models with node attributes for graph analysis Bayesian random graph models [Miller et al., 2009, Zhou, 2015, Caron and Fox, 2017] have been successfully used in relational graph analysis on the tasks of community detection and link prediction. However, many existing models rely on the assumption that the majority of the links of a graph are observed, which is usually unfeasible in practice. How can we obtain good community detection and link prediction performance when only a tiny proportion of the links are observed? This research answers this question by developing models that leverage node attributes, such as user profiles of a social network and author research interests in a bibliographic graph. In the research of Zhao et al. [2017a], an effective Bayesian random graph model is proposed, which regresses a node's latent representations on its attributes, capturing the effect that nodes with similar attributes are likely to be assigned to same communities. The elaborated model structure also facilitates an efficient learning algorithm that utilises the sparsity of both graphs and node attributes. The proposed model achieves the state-of-the-art link prediction results, especially with highly incomplete relational graphs. Besides graph analysis, the proposed "regression to latent representation" idea has been adapted in many other areas such as text analysis [Zhao et al., 2017c,b, 2018a] in order to improve the performance in the case where observed data are highly incomplete.

Discovering latent topics with meta-data for text analysis An important application of BLFMs is text analysis, where latent factors can be interpreted as distributions over vocabulary words, known as "topics." Conventional latent factor models for texts (i.e., topic models) such as Latent Dirichlet Allocation (LDA) [Blei et al., 2003] learn topics purely from the content of a text corpus, ignoring the meta-data associated with documents like labels, authors, timestamps, and words like word embeddings. This research [Zhao et al., 2017c,b, 2018a] answers the question of how to incorporate such meta-data into the learning of topics so as to get better modelling performance and interpretability. In the work of Zhao et al. [2017c, 2018a], a general topic modelling framework is proposed, which efficiently incorporates both document-level and word-level meta-data in binary form. The intuition of this work is that documents with similar meta-data are likely to discuss similar topics and words having similar meanings (encoded in word embeddings) but different morphological forms are likely to be assigned to the same topics. For example, words like "dog" and "puppy", are likely to be in the same topic, even if they barely co-occur in the corpus. The proposed model achieves significantly better modelling results and interpretability, especially on short texts such as tweets and news headlines, where meta-data play a more significant role. This framework is well-engineered on MALLET[1], which is able to run efficiently with multiple threads in multi-core machines on large-scale datasets. In addition, a focused topic model [Zhao et al., 2017b] is proposed, capturing the effect that most topics are about a specific concept that can be described by a few keywords. Therefore, instead of letting a topic be a distribution of all the vocabulary words, the model allows it to focus on a subset of words and the focusing of topics are informed by external word embeddings. In this work, besides better performance, the contribution of this work includes the idea of encoding the semantics of topics into the same space of word embeddings, which is an elegant solution of capturing out-of-vocabulary words.

Topic structure learning for text understanding Conventional topic models assume topics are independent, which is an unnatural assumption in many text corpora. To address this limitation, several advances in topic modelling have started exploring the semantic correlations of topics, which is referred to as the topic structure learning problem, such as the well-known Correlated Topic Model (CTM) [Lafferty and Blei, 2006] and nested Chinese Restaurant Process (nCRP) [Blei et al., 2010]. One of the major themes in this thesis is on discovering interpretable topic structures to obtain better understandings of texts. In the work of Zhao et al. [2018b], a flexible module is proposed to discover three-structured topic hierarchies with a novel angle from previous ones, which is compatible with many other advanced topic models. In the work of Zhao et al. [2018c], by going beyond the conventional assumption that topics are semantically indivisible, the proposed model discovers the fine-grained semantic structures (named "sub-topics") inside an individual topic with the help of word embeddings. To my knowledge, this is the first work that discovers and solves the sub-topic problem in topic modelling. The proposed approaches enjoy not only better modelling performance on perplexity, topic quality, and downstream applications like document classification, but also fantastic interpretability.

Sparse Bayesian factor models for multi-label learning Multi-label learning [Gibaja and Ventura, 2015, Prabhu and Varma, 2014, Jain et al., 2016, Babbar and Schölkopf, 2017] refers to as the problem of learning to assign a subset of relevant labels to each data sample according to its features, given a large set of candidate labels. Each sample is thus associated with a binary label vector, which denotes the presences/absences of the candidate labels. Multi-label learning problems are ubiquitous in a wide variety of applications, such as image/document tagging, recommender system, and ad-placement. Sparsity is a key property in multi-label learning. Specifically, the dimension of the labels can be extremely large, such as millions in many datasets[2], while most of the samples only have a tiny subset of the labels being active, resulting in that label vectors are usually very sparse. If we consider binary features of samples, the feature vector of an object can also be very sparse given a high dimensional feature space. In the work of Zhao et al. [2018d], a model leveraging the sparsity of both label vectors and binary feature vectors is developed, which leads to a very efficient learning algorithm for multi-label learning. In addition, by utilising the label co-occurrence information, the proposed model yields improved prediction accuracies, especially in the case where there is a significant fraction of missing labels.

[1] `http://mallet.cs.umass.edu`

[2] `http://manikvarma.org/downloads/XC/XMLRepository.html`

1.3 Thesis Outline

The remaining parts of this thesis are outlined as follows.

Chapter 2 covers the fundamentals of Bayesian analysis, which offers necessary background knowledge for subsequent chapters. This chapter presents the building blocks for Bayesian modelling including the choices of data and prior distributions, conjugate priors, and data augmentation techniques used in this thesis. Moreover, this chapter presents the unified framework of Bayesian latent factor models, which will be extended in the models described in the following chapters. In addition, the basics of Bayesian inference for learning Bayesian models are reviewed as well.

Chapter 3 gives a comprehensive review of the related works of Bayesian latent factor models in the areas of text analysis, graph analysis, and multi-label learning. Specifically, for text analysis, various topic models are covered especially for ones with meta-data, and specialising in modelling short texts, and with deep/neural structures. For graph analysis, the Bayesian random graph models based on stochastic block models are mainly covered. For multi-label learning problems, Bayesian methods based on low-dimensional factorisations are mainly discussed.

In Chapter 4, I will present the proposed Bayesian random graph model of Zhao et al. [2017a], which incorporates node attributes to improve the performance of link prediction for graph analysis, especially for the case where a large proportion of links in a graph are unobserved.

In Chapter 5, I will present the proposed topic models that are able to incorporate various meta-data such as document labels and word embeddings [Zhao et al., 2017c, 2018a, 2017b]. These models have obtained the state-of-the-art performance especially for short-text modelling with excellent interpretability.

In Chapter 6, the proposed topic models will be presented, which are able to learn tree-structured topic hierarchies [Zhao et al., 2018b] and discover fine-grained sub-topics with word embeddings [Zhao et al., 2018c], respectively, serving as useful tools for intuitively text understanding.

In Chapter 7, I will introduce the proposed Bayesian latent factor model for the multi-label problem, which is able to leverage the sparsity of features and labels to facilitate efficient learning of the model.

Finally, in Chapter 8, I will summarise the content of this thesis, show systematic comparisons across the proposed models, re-summarise the contributions of my PhD research, and discuss the potential research directions.

Chapter 2

Background Knowledge of Bayesian Analysis

This chapter provides a brief review of the background and some of the advanced techniques of Bayesian analysis, used in this thesis. The notations used in this and the following chapter (Chapter 3) are listed and described in Table 2.1.

2.1 Overview of Bayesian Analysis

Here the term "Bayesian Analysis" (BA) is used to describe the process of building and learning Bayesian models for data analysis. In many applications, we usually assume that data are independent and identically distributed (iid) random variables and Bayesian models can be used to learn the unknown probabilistic distribution that generates those random variables.

Suppose a data sample we would like to analyse is $\boldsymbol{x}$, which is assumed to be generated from an unknown distribution parametrised by $\boldsymbol{\theta}$: $\boldsymbol{x} \sim \mathrm{p}(\boldsymbol{x} \mid \boldsymbol{\theta})$ referred to as the *data distribution* or *data likelihood*.

Given a collection of data samples denoted as $\mathbf{X}$, the task of BA is to estimate the unknown parameter $\boldsymbol{\theta}$, from the *posterior distribution*, $\mathrm{p}(\boldsymbol{\theta} \mid \mathbf{X})$. Instead of searching for the "right" $\boldsymbol{\theta}$ without any constraints, BA usually treats $\boldsymbol{\theta}$ as a latent random variable as well, drawn from the *prior distribution*: $\boldsymbol{\theta} \sim \mathrm{p}(\boldsymbol{\theta} \mid \boldsymbol{\alpha})$. For the parameter of the prior distribution, $\boldsymbol{\alpha}$, one can either treat it as a hyper-parameter of the model or further impose another prior distribution on top of it. By stacking priors on top of priors, we are able to build *hierarchical Bayesian models*.

With Bayes' theorem, the posterior distribution can be computed as:

$$\mathrm{p}(\boldsymbol{\theta} \mid \mathbf{X}, \boldsymbol{\alpha}) = \frac{\mathrm{p}(\mathbf{X} \mid \boldsymbol{\theta})\, \mathrm{p}(\boldsymbol{\theta} \mid \boldsymbol{\alpha})}{\mathrm{p}(\mathbf{X} \mid \boldsymbol{\alpha})} = \frac{\mathrm{p}(\mathbf{X} \mid \boldsymbol{\theta})\, \mathrm{p}(\boldsymbol{\theta} \mid \boldsymbol{\alpha})}{\int \mathrm{p}(\mathbf{X} \mid \boldsymbol{\theta})\, \mathrm{p}(\boldsymbol{\theta} \mid \boldsymbol{\alpha}) \mathrm{d}\boldsymbol{\theta}}, \tag{2.1}$$

where $\mathrm{p}(\mathbf{X} \mid \boldsymbol{\alpha})$ is the *marginal distribution* with the parameter $\boldsymbol{\theta}$ marginalised/integrated out.

After the model parameters are estimated, an important use of a Bayesian model is to predict new samples according to the *predictive distribution*:

$$\mathrm{p}(\boldsymbol{x}^* \mid \mathbf{X}) = \int \mathrm{p}(\boldsymbol{x}^* \mid \boldsymbol{\theta})\, \mathrm{p}(\boldsymbol{\theta} \mid \mathbf{X}) \mathrm{d}\boldsymbol{\theta}, \tag{2.2}$$

where the integral is usually hard to compute and it is common to use Monte Carlo integration of $\boldsymbol{\theta}$ with S samples $\{\boldsymbol{\theta}^{<s>}\}_{1,S}$ sampled from $\mathrm{p}(\boldsymbol{\theta} \mid \mathbf{X})$, detailed as follows:

$$\mathrm{p}(\boldsymbol{x}^* \mid \mathbf{X}) \approx \sum_{s=1}^{S} \mathrm{p}(\boldsymbol{x}^* \mid \boldsymbol{\theta}^{<s>})/S. \tag{2.3}$$

Table 2.1: List of notations used in Chapter 2 and 3.

Type	Notation	Description
Vector/Matrix	$\mathcal{X}$	Collection of objects
	$\boldsymbol{x} \in \{0,1\}^V$	V dimensional binary vector
	$\boldsymbol{x} \in \mathbb{N}^V$	V dimensional count vector
	$\boldsymbol{x} \in \mathbb{R}^V$	V dimensional real-valued vector
	$\boldsymbol{x} \in \mathbb{R}_+^V$	V dimensional real-valued non-negative vector
	x_v	v^{th} item of $\boldsymbol{x}$
	$\boldsymbol{x}_{\neg v}$	$V-1$ dimensional vector excluding the v^{th} item
	$x_{\cdot}$	$\sum_{v=1}^{V} x_v$
	$\mathbf{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N]$	V by N matrix, the i^{th} of which is $\boldsymbol{x}_i$
	x_{ij}	Item on the i^{th} row and j^{th} column
	$\boldsymbol{x}^\top$, $\mathbf{X}^\top$	Transpose of vector and matrix
	w_i^j, z_i^j	j^{th} word in document i and its topic
	$\boldsymbol{\theta}^{<s>}$	Model parameters obtained from the s^{th} sample
	$\boldsymbol{\theta}^{(s)}$	Vector in the s^{th} layer in a hierarchical model
	$\boldsymbol{x}^*$	V dimensional vector presenting a testing sample
Bayesian statistics	$\mathrm{p}(\cdot)$	Probability distribution
	$\mathrm{p}(\cdot \mid \cdot)$	Conditional probability distribution
	$\overset{\text{iid}}{\sim}$	Independent and identically distributed
	Ω	Complete separable metric space
	S_t^n	Stirling number of the first kind
Function	$\Gamma(\cdot)$	Gamma function
	$\delta(\cdot)$	Dirac delta function
	$\sigma(\cdot)$	Logistic function
	$\mathrm{I}(\cdot)$	Indicator function

Selecting data and prior distributions, and constructing the connections of those distributions are the main tasks of *Bayesian modelling*, which is the first step of BA. After a model is built, the learning of the model is about the inference of the latent variables from their posterior distributions, referred to as *Bayesian inference*. Moreover, the process of drawing samples of the latent variables/model parameters from their prior distributions down to a data sample is referred to as the *generative process* of a model, while given the data samples, the inference of the latent variables/model parameters from their posterior distributions, which inverts the generative process, is referred to as the *inference process* of the model.

In the following sections, the techniques of Bayesian modelling and inference highly related to this thesis will be elaborated on.

2.2 Bayesian Modelling

The primary task of BA is to approximate the true unknown data distribution. One feasible way of doing this is to assume the data samples are generated from a known prescribed probabilistic distribution with unknown parameters, expecting that this distribution is able to capture the properties of the data and give a good approximation to the true data distribution by adjusting its parameters. The benefit of this is that these distributions have been heavily-studied in the literature and the sampling and inference algorithms have been well-engineered in many programming languages and on many platforms. Similarly, when building hierarchical models, we can impose prescribed distributions on the latent variables as their prior. Therefore, the task of Bayesian modelling can be presented as choosing properly prescribed distributions for the data and latent variables and connecting them with proper structures[1].

In this section, several commonly-used probability distributions are presented, which are building blocks for hierarchical Bayesian models in this research.

2.2.1 Probability Distributions for Discrete Data

As my research focuses on Bayesian modelling and inference for discrete data, I will mainly present the distributions that can be used as data distributions for discrete data. Suppose that the observations of a data sample is V dimensional vector, which can either be binary $\boldsymbol{x} \in \{0,1\}^V$ or count-valued $\boldsymbol{x} \in \mathbb{N}^V$, where $\mathbb{N} = \{0,1,2,\cdots\}$. The data collection consists of N independent and identically distributed (iid) samples, denoted as $\mathbf{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N]$, meaning that $\mathbf{X}$ is a V by N discrete matrix and $\boldsymbol{x}_i$ is its i^{th} column.

If $\boldsymbol{x}$ is a binary vector, a common choice is to generate each element of it from the Bernoulli distribution[2].

$$\boldsymbol{x} \sim \text{Bern}(\boldsymbol{\theta}), \tag{2.4}$$

where $\boldsymbol{\theta} \in (0,1)^V$.

In a special case of binary data sample, where each sample has only one active observation, one can generate it from the categorical distribution with the one-hot presentation:

$$\boldsymbol{x} \sim \text{Cat}(\boldsymbol{\theta}), \tag{2.5}$$

where $\boldsymbol{\theta}$ is a probability vector, i.e., $\sum_{v=1}^{V} \theta_v = 1$.

If $\boldsymbol{x}$ is a count-valued vector, one can use the Poisson, negative-binomial, or multinomial distributions as the data distribution.

$$\boldsymbol{x} \sim \text{Pois}(\boldsymbol{\theta}), \tag{2.6}$$

$$\boldsymbol{x} \sim \text{NB}(\boldsymbol{r}, \boldsymbol{p}), \tag{2.7}$$

$$\boldsymbol{x} \sim \text{Multi}(x_\cdot, \boldsymbol{\theta}), \tag{2.8}$$

where for Poisson, $\boldsymbol{\theta} \in \mathbb{R}_+^V$ and $\mathbb{R}_+ = \{x : x \geq 0\}$; for negative-binomial, $\boldsymbol{r} \in \mathbb{R}_+^V$ and $\boldsymbol{p} \in (0,1)^V$; for multinomial, $\boldsymbol{\theta}$ is a probability vector; and $x_\cdot = \sum_{v=1}^{V} x_v$[3].

The comparisons between the above three choices can be summarised as follows:

- For Poisson and negative-binomial distributions, each element of $\boldsymbol{x}$ is individually generated, i.e. $x_v \sim \text{Pois}(\theta_v)$ and $x_v \sim \text{NB}(r_v, p_v)$. While for multinomial, which is originally a multivariate distribution, all the elements of $\boldsymbol{x}$ are jointly generated.
- When the total count of a data vector $x_\cdot$ is known, $\boldsymbol{x} \sim \text{Pois}(\boldsymbol{\theta})$ is equivalent to $\boldsymbol{x} \sim \text{Multi}(x_\cdot, \boldsymbol{\theta}/\theta_\cdot)$.
- With only one free parameter, the Poisson distribution does not allow the variance to be adjusted independently from the mean, while the negative-binomial distribution consist of one more parameter to model the data variance independently. This property of the negative-binomial distribution is important to capture *overdispersion*, which usually exists in large-scale, high-dimensional, sparse, discrete data.

[1] Note that recently, distributions constructed by nonlinear transformations of random noise with deep neural networks have been used in deep generative models [Kingma and Welling, 2013, Goodfellow et al., 2014], which are no longer prescribed distributions. The discussion on those models is beyond the scope of this thesis.

[2] Hereafter, $x \sim \text{p}(x \mid \alpha)$ is used to note a data sample drawn from a distribution parametrised by α. Specifically, x could be a scalar or a vector. If x is a V-dimensional vector, then we have $\boldsymbol{x} \sim \text{p}(\boldsymbol{x} \mid \boldsymbol{\alpha})$, where $\text{p}(\boldsymbol{x} \mid \boldsymbol{\alpha})$ can be a multivariate distribution (e.g., Dirichlet) or a univariate distribution (e.g., gamma). The latter case means that each dimension of $\boldsymbol{x}$, x_v, is independently generated from the univariate distribution with parameter α_v. Note that in the case of $\boldsymbol{\alpha} = [\alpha_0, \cdots, \alpha_0]$, I will denote $\boldsymbol{x} \sim \text{p}(\boldsymbol{x} \mid \alpha_0)$ for convenience.

[3] Hereafter, I will use $\cdot$ to denote the summation over the elements of a vector or the elements in a specific dimension of a matrix.

Table 2.2: Conjugate priors.

Data distribution	Prior distribution	Posterior distribution
$x \sim \mathrm{Bern}(\theta)$	$\theta \sim \mathrm{Beta}(\alpha, 1/\beta)$	$\theta \sim \mathrm{Beta}(\alpha + x, \beta + 1 - x)$
$x \sim \mathrm{Pois}(\theta)$	$\theta \sim \mathrm{Gamma}(\alpha, 1/\beta)$[4]	$\theta \sim \mathrm{Gamma}(\alpha + x, 1/(\beta + 1))$
$x \sim \mathrm{NB}(r, p)$	$p \sim \mathrm{Beta}(\alpha, \beta)$	$p \sim \mathrm{Beta}(\alpha + x, \beta + r)$
$\boldsymbol{x} \sim \mathrm{Cat}(\boldsymbol{\theta})$	$\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha})$	$\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha} + \boldsymbol{x})$
$\boldsymbol{x} \sim \mathrm{Multi}(x., \boldsymbol{\theta})$	$\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha})$	$\boldsymbol{\theta} \sim \mathrm{Dir}(\boldsymbol{\alpha} + \boldsymbol{x})$
$x \sim \mathrm{Gamma}(\alpha, 1/\beta)$	$\beta \sim \mathrm{Gamma}(\alpha_0, 1/\beta_0)$	$\beta \sim \mathrm{Gamma}(\alpha_0 + \alpha, 1/(\beta_0 + x))$

2.2.2 Conjugate Prior Distributions

After presenting the choices of data distributions for discrete data, here I present the choices of prior distributions. Theoretically, one can use any probability measure as the prior distribution of a latent variable, as long as it satisfies the constraints, e.g., the parameters of a multinomial need to be normalised. In practice, people usually take the following two factors into consideration: whether the prior distributions capture the characteristics of the data and incorporate our prior knowledge, and whether they facilitate the convenience of inference. For the latter, a common choice is *conjugate priors*. Conjugate priors enable the posterior to have the same algebraic form as the prior, which significantly reduces the complexity of inference by bypassing the computation of the integral in Eq. (2.1). With conjugacy, the posterior can also be presented as a prescribed distribution, which can easily be sampled. Moreover, conjugate priors also give intuitive demonstrations on how a likelihood function updates a prior distribution. Therefore, conjugate priors play an important role in Bayesian inference and are heavily used in this research.

Several commonly-used conjugate priors of my research are listed in Table 2.2 and take the conjugacy of the Bernoulli and beta as an example to further demonstrate conjugate priors as follows: Before seeing any data, we assume that the Bernoulli parameter should be drawn from the prior distribution, $\theta \sim \mathrm{Beta}(\alpha, \beta)$. After observing one sample x, we can update the model parameter by the posterior, $\theta \sim \mathrm{Beta}(\alpha + x, \beta + 1 - x)$, which is a beta distribution as well. In this example, we can clearly see how a data sample updates the model parameter with Bayes' theorem. It can also be observed that the posterior is affected by both the prior and the data, where if fewer samples are observed or the data space is sparse, the prior would have a stronger influence on the posterior, while the uncertainty of the posterior is reduced when more samples are observed.

Now, some important properties of the distributions and relationships between them are presented, which are frequently-used in this research.

- If a distribution $\mathrm{p}(x \mid \theta)$ satisfies that $x_1 \sim \mathrm{p}(x \mid \theta_1)$ and $x_2 \sim \mathrm{p}(x \mid \theta_2)$, it has the *summation* property. In the above distributions, summation property exists in Poisson and gamma (with the same scale parameter).

- The gamma distribution satisfies the *scaling* property, meaning that if $x \sim \mathrm{Gamma}(\alpha, 1/\beta)$, then $cx \sim \mathrm{Gamma}(\alpha, 1/(c\beta))$.

- The Dirichlet distribution has the *aggregation* property, meaning that if $(x_1, \cdots, x_V) \sim \mathrm{Dir}(\alpha_1, \cdots, \alpha_V)$, then $(x_1, \cdots, x_i + x_j, \cdots, x_V) \sim \mathrm{Dir}(\alpha_1, \cdots, \alpha_i + \alpha_j, \cdots, \alpha_V)$.

- Suppose $(x_1, \cdots, x_V) \sim \mathrm{Dir}(\alpha_1, \cdots, \alpha_V)$, it is equivalent to draw $y_i \sim \mathrm{Gamma}(\alpha_i, 1/\beta)$ $(1 \leq i \leq V)$ and then normalise it: $x_i = \frac{y_i}{\sum_{i=1}^{V} y_i}$. That is to say, normalising a

[4] The two parameters are the shape and scale parameters of gamma, respectively.

vector of gamma distributed variables (with the same scale) ends up with a vector of Dirichlet variables. Poisson and multinomial have a similar property: $y_1 \sim \text{Pois}(\alpha_1), \cdots, y_V \sim \text{Pois}(\alpha_V)$ is equivalent to $(x_1, \cdots, x_V) \sim \text{Multi}(x., \alpha_1/\alpha., \cdots, \alpha_V/\alpha.)$.

- The negative-binomial distribution can be viewed as a continuous mixture of Poisson distributions, where the mixing weight is drawn from a gamma distribution, i.e., $x \sim \text{NB}(r, p)$ is equivalent to $x \sim \int_{\theta\sim\text{Gamma}(r,\frac{1-p}{p})} \text{Pois}(\theta)\theta \text{d}\theta$.

Note that for the above distributions with more than one set of parameters, e.g. gamma and negative-binomial, there are no trivial conjugate priors for both of the parameters. However, with techniques such as data augmentation, one can relax this constraint and obtain more flexibility of using conjugate priors, which I will elaborate on in the following section.

2.2.3 Data Augmentation for Non-Conjugate Priors

Conjugacy gives us inference convenience and intuitive interpretations, but it also limits the flexibility of building Bayesian models. To relax this limitation, I present the technique called *data augmentation* for non-conjugate model constructions. Specifically, suppose we would like to build a model like $x \sim \text{p}(x \mid \alpha)$ and $\alpha \sim \text{p}(\alpha \mid \beta)$, and $\text{p}(x \mid \alpha)$ is not conjugate to $\text{p}(\alpha \mid \beta)$. One can introduce an auxiliary latent variable l such that:

$$\text{p}(l, x \mid \alpha) = \text{p}(l \mid x, \alpha)\,\text{p}(x \mid \alpha) = \text{p}(x \mid l, \alpha)\,\text{p}(l \mid \alpha). \tag{2.9}$$

With carefully choosing the prior distribution of l so that we can easily sample it from $l \sim \text{p}(l \mid x, \alpha)$, we may be able to get the conjugacy between $\text{p}(l \mid \alpha)$ and $\text{p}(\alpha \mid \beta)$ so as to sample α conditioned on l and β. This is the basic idea of data augmentation. Next, I elaborate on the details of data augmentation techniques used in my research.

Data Augmentation for Poisson-Gamma-Gamma Models

Now we consider the following hierarchical Bayesian model:

$$\alpha \sim \text{Gamma}(a_0, 1/b_0), \tag{2.10}$$
$$\beta \sim \text{Gamma}(c_0, 1/d_0), \tag{2.11}$$
$$\theta \sim \text{Gamma}(\alpha, 1/\beta), \tag{2.12}$$
$$x \sim \text{Pois}(\theta). \tag{2.13}$$

Given the Poisson-Gamma conjugacy in Table 2.2, we can show that θ's posterior is also a gamma: $\theta \sim \text{Gamma}\left(\alpha + x, 1/(\beta + 1)\right)$. Note that this gamma posterior is conjugate to the gamma prior of β, which is the scale parameter. Therefore, we can easily figure out that the posterior of β is a gamma distribution as well: $\beta \sim \text{Gamma}(c_0 + \alpha, 1/(d_0 + \theta))$.

Now conditioned on α and β, if we marginalise θ out, we can get:

$$\text{p}(x \mid \alpha) \propto \frac{\Gamma(\alpha + x)}{\Gamma(\alpha)} \left(\frac{\beta}{\beta + 1}\right)^{\alpha}, \tag{2.14}$$

which is not conjugate to the gamma prior of α.

The challenge here is the gamma ratio in the above equation, $\frac{\Gamma(\alpha+x)}{\Gamma(\alpha)}$, which is also the Pochhammer symbol for a rising factorial. Fortunately, it can be augmented with an auxiliary variable t: $\frac{\Gamma(\alpha+t)}{\Gamma(\alpha)} = \sum_{l=0}^{x} S_t^x \alpha^t$ where S_t^x indicates an unsigned Stirling number of the first kind [Chen et al., 2011, Teh et al., 2012, Zhou and Carin, 2015]. With t, Eq. (2.14) can be augmented as:

$$\text{p}(t \mid \alpha) \propto \alpha^t \left(\frac{\beta}{\beta + 1}\right)^{\alpha} = \alpha^t e^{-\log\frac{\beta+1}{\beta}\alpha}, \tag{2.15}$$

which is a gamma-like likelihood and conjugate to $\mathrm{p}(\alpha \mid a_0, b_0)$.

Therefore, the posterior of α can be written as:

$$\alpha \sim \text{Gamma}\left(a_0 + t, 1/\left(b_0 + \log \frac{\beta}{\beta+1}\right)\right). \tag{2.16}$$

Now we need to sample t from $t \sim \mathrm{p}(t \mid x, \alpha)$. Fortunately, the above gamma ratio is the normalisation term of the posterior of a Chinese Restaurant Process (detailed in Section 2.2.4) with α as its concentration parameter, x as the number of customers, and t as the number of tables assigned for those customers. If we only care about the number of tables, i.e., t regardless of the specific sitting arrangement of an individual customer, we can sample t by:

$$t \sim \sum_{i=1}^{x} \text{Bern}\left(\frac{\alpha}{\alpha + i - 1}\right), \tag{2.17}$$

where $\frac{\alpha}{\alpha+i-1}$ is the probability of opening a new table for the i^{th} customer. The distribution of t is called the Chinese Restaurant Table (CRT) distribution by Zhou and Carin [2015]. Adopting this name, this augmentation is referred to as the *CRT augmentation*.

This CRT augmentation technique enables us to build hierarchical models with gamma distributions for Poisson-distributed data such as in Zhou et al. [2012b], Hu et al. [2016a], Zhou et al. [2016] and also my research [Zhao et al., 2018a, 2017c, 2018d]. For example, if we impose another gamma prior on a_0, the same augmentation can be applied to obtain the gamma posterior of a_0. Therefore, multi-layer models can be built by stacking gamma priors.

Data Augmentation for Multinomial-Dirichlet-Gamma Models

Now I show another data augmentation technique that is related to the CRT one but is used in the following model:

$$\alpha_v \sim \text{Gamma}(a_0, 1/b_0), \tag{2.18}$$

$$\boldsymbol{\theta} \sim \text{Dir}(\boldsymbol{\alpha}), \tag{2.19}$$

$$\boldsymbol{x} \sim \text{Multi}(x_\cdot, \boldsymbol{\theta}), \tag{2.20}$$

where $\boldsymbol{x} \in \mathbb{N}^V$, $\boldsymbol{\theta}$ is a probability vector, and $\boldsymbol{\alpha} \in \mathbb{R}_+^V$.

Given the Multinomial-Dirichlet conjugacy in Table 2.2, we can show that θ's posterior is also a Dirichlet:

$$\boldsymbol{\theta} \sim \text{Dir}(\alpha + \boldsymbol{x}). \tag{2.21}$$

If we marginalise $\boldsymbol{\theta}$ out, we can get:

$$\mathrm{p}(\boldsymbol{x} \mid \boldsymbol{\alpha}) \propto \frac{\Gamma(\alpha_v)}{\Gamma(\alpha_\cdot + x_\cdot)} \prod_{v=1}^{V} \frac{\Gamma(\alpha_v + x_v)}{\Gamma(\alpha_v)}. \tag{2.22}$$

Note that the CRT augmentation helps us deal with the left-hand side (LHS) gamma ratio. While for the right-hand side (RHS) one, given the fact that this gamma ratio is a normalisation term of a beta distribution, i.e., $\text{Beta}(\alpha_\cdot, x_\cdot)$, we can introduce a beta-distributed auxiliary variable p and augment the RHS gamma ratio as follows:

$$\frac{\Gamma(\alpha_v)}{\Gamma(\alpha_\cdot + x_\cdot)} \propto \int_p p^{\alpha_\cdot - 1}(1-p)^{x_\cdot - 1}. \tag{2.23}$$

Together with the CRT augmentation (for each v, an auxiliary CRT variable t_v is introduced), we can write Eq. (2.23) as:

$$\mathrm{p}(\boldsymbol{t}, p \mid \boldsymbol{\alpha}) \propto \prod_{v=1}^{V} q^{\alpha_v} \alpha_v^{t_v} = \prod_{v=1}^{V} \alpha_v^{t_v} e^{-\log \frac{1}{q} \alpha_v}, \tag{2.24}$$

which is conjugate to the gamma prior of α_v. Now the posterior of α_v is:

$$\alpha_v \sim \text{Gamma}\left(a_0 + t_v, 1/\left(b_0 + \log \frac{1}{q}\right)\right), \tag{2.25}$$

where $q \sim \text{Beta}(\alpha_\cdot, x_\cdot)$.

According to the distributions of the auxiliary variables, this technique is referred to as the beta-CRT augmentation. Note that in addition to beta-CRT, the CRT augmentation can also be applied to model a vector of count-valued data:

$$\alpha_v \sim \text{Gamma}(a_0, 1/b_0), \tag{2.26}$$
$$\beta \sim \text{Gamma}(c_0, 1/d_0), \tag{2.27}$$
$$\boldsymbol{\theta} \sim \text{Gamma}(\boldsymbol{\alpha}, 1/\beta), \tag{2.28}$$
$$\boldsymbol{x} \sim \text{Pois}(\boldsymbol{\theta}), \tag{2.29}$$

where $\boldsymbol{x} \in \mathbb{N}^V$.

The major difference between CRT and beta-CRT for modelling a data vector is that in the former one, $\boldsymbol{\theta}$ is unnormalised, while it is normalised probability vector in the latter one. In some cases, normalised probability vectors are required to avoid the unidentifiable issue. The beta-CRT augmentation has been heavily-used in building hierarchical models based on multinomial-Dirichlet of my research, such as in Zhao et al. [2017c, 2018a,c].

Data Augmentation with Pólya-Gamma Auxiliary Variables

The final augmentation technique I present here is the *Pólya-Gamma augmentation* proposed by Polson et al. [2013] for the following model:

$$\frac{\left(e^{\psi}\right)^a}{(1 + e^{\psi})^b} = 2^{-b} e^{\mathcal{K}\psi} \int_0^\infty e^{-\omega\psi^2/2} \mathrm{p}(\omega) \mathrm{d}\omega, \tag{2.30}$$

where $b > 0$, $\mathcal{K} = a - b/2$, and the prior distribution of ω is $\omega \sim \text{PG}(b, 0)$, and PG denotes the Pólya-Gamma distribution.

With this augmentation, the LHS binomial-like likelihood is augmented into a Gaussian-like likelihood, which enables us to impose a Gaussian prior on ψ. Shown in Polson et al. [2013], ω has a Pólya-Gamma posterior as well: $\omega \sim \text{PG}(b, \psi)$[5].

Originally, the Pólya-Gamma augmentation is used for logistic regression models such as:

$$y \sim \text{Bern}(\sigma(\boldsymbol{x}\boldsymbol{\beta}^\top)), \tag{2.31}$$

where y is the label of an object with features as $\boldsymbol{x} \in \mathbb{R}^V$, $\sigma(a) = \frac{1}{1+e^{-a}}$ is the logistic function, and $\boldsymbol{\beta} \in \mathbb{R}^V$ can be imposed with a normal prior.

The technique can be generalised into negative-binomial regression [Zhou et al., 2012b], which is a simplified construction of my model of incorporating word embeddings for topic modelling [Zhao et al., 2018c], shown as follows:

$$\theta \sim \text{Gamma}(a_0, e^{\boldsymbol{x}\boldsymbol{\beta}^\top}), \tag{2.32}$$
$$y \sim \text{Pois}(\theta). \tag{2.33}$$

[5] Efficient sampling algorithms for PG can be found in Zhou [2018].

If we marginalise out θ in the above model, we can obtain:

$$\mathrm{p}(y \mid \beta, a_0) \propto \frac{(e^{x\beta^\top})^y}{(1+e^{x\beta^\top})^{y+a_0}}, \tag{2.34}$$

where the Pólya-Gamma augmentation can be used then.

2.2.4 Stochastic Processes

In addition to probability distributions, stochastic processes are important building blocks for *Bayesian nonparametric models*. A stochastic process can be defined as a random variable collection that is indexed by some mathematical set, that is to say, each random variable of the stochastic process is uniquely associated with an element in the set. A Bayesian nonparametric model is a probabilistic model built with stochastic processes, whose model structure grows in the size with the amount of data.

In addition to probability distributions, stochastic processes are important building blocks for *Bayesian nonparametric models*. A stochastic process can be defined, informally, as a collection of random variables that are indexed by some mathematical set. That is to say, each random variable of the stochastic process is uniquely associated with an element in the set. For instance, the indexing set may be positive integers, then the stochastic process represents a sequence of random variables. The indexing set may be the 2-D real plane, then the stochastic process represents a distribution on a "field." Stochastic processes are often constructed in mathematical statistics using Poisson processes. If the indexing set for the process is the set Ω, then an inhomogeneous Poisson process on Ω can be used to define where the impulses on Ω are, i.e., which countable subset of Ω is present in the sample of the process. Alternatively one can place an inhomogeneous Poisson process on $\mathcal{R}_+ \times \Omega$ so that one gets an extra weight $\in \mathcal{R}_+$ (a positive real number) drawn as well. In this case, the sample from a stochastic process is a countable set of points $(w,t) \in \mathcal{R}_+ \times \Omega$, and the Poisson process is referred to as a Lévy measure. Manipulation of these yields a rich set of methods for Bayesian non-parametric theory [Lijoi and Prünster, 2010].

A Bayesian nonparametric model is a probabilistic model built with stochastic processes. This is typically done so that the model structure grows in size with the amount of data. In general, Bayesian nonparametrics can be a rather broad area, but I limit my discussion of stochastic processes from an application perspective and especially focus on gamma processes and Dirichlet processes, which are highly related to my research.

Gamma Processes

Following the description of Zhou [2015], a gamma process [Ferguson, 1973, Wolpert et al., 2011] can be defined on a product space $\mathbb{R}_+ \times \Omega$ as:

$$G \sim \Gamma\mathrm{P}(G_0, 1/c_0), \tag{2.35}$$

where Ω is a complete separable metric space, c_0 is the concentration parameter, G_0 is a finite and continuous base measure over Ω, such that $G(A_i) \sim \mathrm{Gamma}(G_0(A_i), 1/c_0)$ are independent gamma variables for a disjoint partition A_i of Ω.

The Lévy measure of the above gamma process can be shown as follows:

$$\nu(drd\phi) = r^{-1}e^{-c_0 r} dr G_0(d\phi). \tag{2.36}$$

Note that the Poisson intensity of the whole product space is infinite: $\nu(\mathbb{R}_+ \times \Omega) = \infty$ while $\int_{\mathbb{R}_+ \times \Omega} r\nu(drd\phi)$ is finite. Therefore, a draw from the gamma process with countably infinite atoms, can be expressed as follows:

$$G_0 = \gamma_0 g_0, \tag{2.37}$$
$$g_0(d\phi) = G_0(d\phi)/\gamma_0, \tag{2.38}$$
$$\phi_k \overset{\text{iid}}{\sim} g_0, \tag{2.39}$$
$$G = \sum_{k=1}^{\infty} r_k \delta(\phi_k), \tag{2.40}$$

where $\gamma_0 = G_0(\Omega)$ is the mass parameter. Note that the number of atoms with weights greater than $\tau \in \mathbb{R}_+$ in the prior follows a Poisson distribution $\text{Pois}(\gamma_0 \int_\tau^{\text{inf}} r^{-1}e^{-cr}\text{d}r)$, the rate of which decreases as τ increases. Therefore, a gamma process based model has an inherent shrinkage mechanism.

In practice, a gamma process with a finite and continuous base measure can be approximated with a hierarchical model with gamma distributions. Specifically, suppose we would like to draw a K dimensional vector from the above gamma process, we can implement it with a truncated model as:

$$\gamma_0 \sim \text{Gamma}(a_0, 1/b_0), \tag{2.41}$$
$$r_k \sim \text{Gamma}(\gamma_0/K, 1/c_0), \tag{2.42}$$

where as $K \to \infty$, the exact gamma process can be recovered. In the implementation, we usually set K (truncation level) to be large enough to obtained a good approximation to the truly infinite model.

Dirichlet Processes

Another important stochastic process is the Dirichlet Process (DP) [Ferguson, 1973]. Recall that in a draw from a gamma process is: $G = \sum_{k=1}^{\infty} r_k \delta(\phi_k)$. By normalising the weights $\pi_k = \frac{r_k}{\sum_{k=1}^{\infty} r_k}$, we can get a process as:

$$D = \sum_{k=1}^{\infty} \pi_k \delta(\phi_k), \tag{2.43}$$

which is known as a DP, denoted as $D \sim \text{DP}(H_0, \alpha_0)$. Here $H_0 = G_0/\alpha_0$ and $\alpha_0 = G_0(\Omega)$ is the concentration parameter.

Intuitively, DP can be viewed as an infinite generalisation of the Dirichlet distribution. Suppose for any measurable partition $(A_1, \cdots, A_K)$ of Ω, $(D(A_1), \cdots, D(A_K))$ is Dirichlet distributed as follows:

$$(D(A_1), \cdots, D(A_K)) \sim \text{Dir}(\alpha_0 H_0(A_1), \cdots, \alpha_0 H_0(A_K)). \tag{2.44}$$

Therefore, if the base measure $H_0 = (h_1, \cdots, h_K)$ is a K dimensional discrete probability vector, then DP becomes a Dirichlet distribution: $\text{Dir}(\alpha_0(h_1, \cdots, h_K))$. On the other hand, if H_0 is a non-discrete distribution, a DP is essentially an infinite dimensional Dirichlet distribution.

To draw a sample from a DP, two well-known constructions of a DP have been proposed: the stick-breaking construction and the Chinese Restaurant Process construction.

Stick-breaking construction According to Sethuraman [1994], the stick-breaking construction of a DP, $D \sim \mathrm{DP}(H_0, \alpha_0)$, is as follows:

$$V_k \sim \mathrm{Beta}(1, \alpha_0), \tag{2.45}$$

$$\phi_k \overset{\text{iid}}{\sim} H_0, \tag{2.46}$$

$$p_k = V_k \prod_{j=1}^{k-1} (1 - V_j), \tag{2.47}$$

$$D = \sum_{k=1}^{\infty} p_k \delta(\phi_k), \tag{2.48}$$

where $\sum_{k=1}^{\infty} p_k = 1$.

As its name implies, the stick-breaking construction can be intuitively understood as a process of breaking a stick with length 1 into pieces: one first breaks the stick into two pieces at p_1 with length V_1 and $1 - V_1$, respectively; then breaks the second piece with length $(1 - V_1)V_2$ at p_2; ...

Chinese restaurant process construction Also known as the Blackwell-Macqueen urn scheme [Blackwell et al., 1973], the Chinese Restaurant Process (CRP) construction asymptotically produces a partition of integers and shows how the samples from a Dirichlet process exhibit a clustering property [Teh et al., 2005].

For a CRP, $C \sim \mathrm{CRP}(H_0, \alpha_0)$, suppose $(x_1, x_2, \cdots)$ are iid samples drawn from it and $(\phi_1, \phi_2, \cdots)$ are distinct values drawn from the discrete base measure H_0. I further denote $n_k = \sum_{i=1}^{N} \mathbf{1}_{x_i = \phi_k}$, where N is the number of samples that have been drawn from the CRP. Metaphorically, one can interpret the samples of a CRP as the customers coming into a restaurant, the samples from the base measure are the table numbers, and n_k is the number of customers assigned to the table number ϕ_k. The CRP generates the sitting arrangements of those customers. Specifically, the next, i.e., $(n + 1)^{\text{th}}$ customer's sitting arrangement is with the following probabilities:

$$\mathrm{p}(\text{the customer seated at table } \phi_k) \propto \frac{n_k}{n_k + \alpha_0}, \tag{2.49}$$

$$\mathrm{p}(\text{the customer seated at a new table}) \propto \frac{\alpha}{N + \alpha_0}. \tag{2.50}$$

From the above process, one can clearly see the clustering property or the so-called "the rich get richer" phenomenon: if more customers are seated in a table, the next customer is more likely to be seated in this table.

2.2.5 General Patterns of Hierarchical Bayesian Models

After introducing the commonly-used building blocks of Bayesian models, here I present several general patterns of constructing hierarchical Bayesian models with those building blocks, although specific model constructions usually depend on data and applications.

Like many other machine learning models such as deep neural networks, one usually imposes prior distributions of a Bayesian model to aggregate the information of data with fewer model parameters and/or incorporate extra knowledge into the model. This research focuses on the data that can be formatted into a discrete matrix $\mathbf{X} \in \mathbb{N}^{V \times N}$, each column of which is the observation (e.g., features or attributes) of an iid data sample, $\boldsymbol{x}_i \in \mathbb{N}^V$.

To analyse such data with a Bayesian model, there are four general patterns for us to use:

- **Single:** $\boldsymbol{x}_i \sim \mathrm{p}(\boldsymbol{x}_i \mid \theta_0)$. This is the simplest model construction with only one parameter, meaning that it ignores the variations between the samples and the

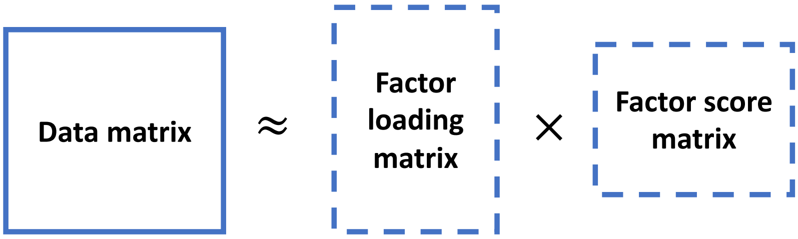


Figure 2.1: Basic framework of BLFMs. The rectangles with solid lines and dash lines are the data matrix and latent matrices, respectively.

features of a sample. When this pattern is directly applied to data, it usually causes underfitting. However, this pattern is often applied to generate the top-layer latent variables of a hierarchical model.

- **Matrix:** $\boldsymbol{x}_i \sim \mathrm{p}(\boldsymbol{x}_i \mid \boldsymbol{\theta}_i)$, $\boldsymbol{\theta}_i$ is a V dimensional vector specific to each sample. This could be an unrealistic pattern for practical applications because the size of the parameter space is the same as the size of the data space. Therefore, no aggregation is conducted.

- **Vector:** $\boldsymbol{x}_i \sim \mathrm{p}(\boldsymbol{x}_i \mid \boldsymbol{\theta}_0)$, $\boldsymbol{\theta}_0$ is a V dimensional vector global to the whole dataset. With this pattern, one actually ignores the variations between samples but captures the variations between features. This pattern can be used in tasks like regression and classification. Moreover, it can also be used to generate the intermediate-layer latent variables of a hierarchical model.

- **Factorisation:** $\boldsymbol{x}_i \sim \mathrm{p}(\boldsymbol{x}_i \mid \boldsymbol{\Phi}\boldsymbol{\theta}_i)$, $\boldsymbol{\theta}_i$ is a K dimensional vector, and $\boldsymbol{\Phi}$ is a V by K matrix. This pattern forms the basic framework of *Bayesian Latent Factor Model* (BLFM), which is the main research theme of this thesis. I will elaborate on the details of this pattern in Section 2.2.6.

With the above patterns as well as the probabilistic distributions, one is able to construct hierarchical Bayesian models for complex data analysis, which is the major research in this thesis.

2.2.6 Bayesian Latent Factor Models

As briefly discussed in Section 2.2.5, Bayesian latent factor models factorise the data matrix $\mathbf{X} \in \mathbb{N}^{V\times N}$ with K latent factors into two latent matrices $\boldsymbol{\Phi} \in \mathbb{R}_+^{V\times K}$ and $\boldsymbol{\Theta} \in \mathbb{R}_+^{K\times N}$. Following the notations of matrix factorisation, $\boldsymbol{\Phi}$ is the *factor loading* matrix, each column of which is a factor encoding the relative importance of each feature; $\boldsymbol{\theta}_i$ is the *factor score* vector of sample i, encoding the relative importance of factor in sample i. As prior distributions are imposed on $\boldsymbol{\theta}_i$ and $\boldsymbol{\Phi}$, a Bayesian model with the above structures is usually called a BLFM, which has extensive applications in machine learning and data analysis. Moreover, as $\boldsymbol{\theta}_i$ is individual to each sample, it is usually called the *local variables* in BFA, while $\boldsymbol{\Phi}$ is the *global variables*. The basic framework of BLFMs is intuitively shown in Figure 2.1.

The most important direction of this thesis research is developing extensions and variations of the above framework of BLFMs for various applications, as well as tackling the inference problems on large-scale data.

2.3 Bayesian Inference

Bayesian inference is the inference procedure of the parameters of a Bayesian model from their posterior distributions. Recall that in Eq. (2.1), the posterior of the model parameter $\boldsymbol{\theta}$ can be obtained by Bayes' theorem, which is normalised by the marginal distribution. Therefore, to exactly compute the posterior, we need to calculate the integral in the marginal distribution, which can be expensive, or usually impossible. Therefore, it calls for approximate Bayesian inference algorithms to evaluate the posterior. Two of the most commonly-used approximate Bayesian inference techniques are *Markov Chain Monte Carlo* (MCMC) sampling and *variational inference*. Instead of directly deriving the posterior distribution, the former approach allows us to obtain samples from the posterior and the posterior can be characterised by the relevant statistics computed from the samples. On the other hand, in the latter approach, the intractable posterior is approximated by the proposed tractable variational distributions, where the approximation is done by minimising a specific distance between the true posterior and the variational distributions.

In general, the principle of *detailed balance* used in MCMC methods ensures that the samples of a Markov chain will eventually converge to the true samples of the posterior. However, in terms of efficiency and scalability, MCMC methods may take many iterations to get converged and it is usually hard to conduct MCMC sampling with batches of data. While for variational methods, the inference problem is transformed into an optimisation one, which can be done by efficient parallel algorithms. However, it is usually tricky to find out how well the posterior is approximated, because the performance can be affected by multiple factors such as the expressiveness of the variational distributions and the local optimums in the optimisation. Recently, extensive research efforts have been devoted to improving the efficiency of MCMC methods and the accuracy of variational inference, such as the approaches of Stochastic Gradient MCMC [Chen et al., 2014] and amortized variational inference [Kingma and Welling, 2013].

In this section, the basics of a specific kind of MCMC sampling methods, called Gibbs sampling, are presented. Gibbs sampling is one of the most widely-used Bayesian inference algorithms and is also primarily used in the later chapters.

2.3.1 Metropolis-Hasting Algorithm

Before going into Gibbs sampling, I briefly present the Metropolis-Hasting (MH) Algorithm [Metropolis et al., 1953, HASTINGS, 1970], which is one of the most important families of MCMC sampling algorithms. Like other MCMC methods, MH sampling is an iterative algorithm as well. Suppose that we would like to sample from a multivariate posterior distribution, $\mathrm{p}(\boldsymbol{\theta} \mid \boldsymbol{x})$, called the *target distribution*. We first need to choose a *proposal distribution* for each dimension of $\boldsymbol{\theta}$, in the form of $\mathrm{g}(\theta_v^* \mid \boldsymbol{\theta}, \boldsymbol{x})$, which is expected to be easy to sample and with similar shape as the target distribution.

In each iteration, for the v^{th} dimension of $\boldsymbol{\theta}$, θ_v, we first sample a new value of it, θ_v^*, from the proposal distribution. We then calculate the ratio of densities for θ_v^* defined as:

$$A = \frac{\mathrm{p}(\theta_v^*, \boldsymbol{\theta}_{\neg v} \mid \boldsymbol{x})\, \mathrm{g}(\theta_v \mid \theta_v^*, \boldsymbol{\theta}_{\neg v}, \boldsymbol{x})}{\mathrm{p}(\boldsymbol{\theta} \mid \boldsymbol{x})\, \mathrm{g}(\theta_v^* \mid \boldsymbol{\theta}, \boldsymbol{x})}, \tag{2.51}$$

where $\boldsymbol{\theta}_{\neg v}$ denotes the dimensions of $\boldsymbol{\theta}$ other than θ_v.

Next, we use $A' = min(A, 1)$ as the *acceptance rate*, which is the probability of accepting θ_v^* as the new value of θ_v. If θ_v^* is not accepted, θ_v retains its current value. The above step is repeated until the sampling is converged.

Compared with Gibbs sampling that I am going to introduce, MH sampling can be more flexible, because conjugacy is not necessarily required. However, it is not always easy to choose the right proposal distribution and the acceptance rate might be very small in some cases, where there could be huge computation wastes when many samples are rejected.

2.3.2 Gibbs Sampling

Gibbs sampling [Geman and Geman, 1984] is a special case of MH sampling, where the conditional posterior distribution of each dimension of θ is defined as the proposal distribution:

$$\mathrm{g}(\theta_v^* \mid \boldsymbol{\theta}, \boldsymbol{x}) = \mathrm{p}(\theta_v^* \mid \boldsymbol{\theta}, \boldsymbol{x}_{\neg v}). \tag{2.52}$$

It can be proved that the acceptance rate of Gibbs sampling with this specification of proposal distribution equals to one, meaning that all candidate values are accepted. Therefore, compared to MH sampling, Gibbs can be more efficient and one can be freed from spending effort on searching for the proposal distribution. The intuition behind Gibbs sampling is that although it is hard to sample directly from the complex full posterior, the conditional posterior may have a simple form, which usually happens in conjugate models. Compared with general MH sampling methods, Gibbs sampling usually requires conjugacy. Therefore, one may need a considerable amount of effort to achieve full/local conjugacy in a model. Section 2.2 provides a detailed introduction to such techniques, which saves the efforts of conducting Gibbs sampling.

In this thesis research, Gibbs sample has been heavily used in the inference of the proposed models, mainly because of the following three reasons:

- As the research theme focuses on developing novel models for data analysis, it is relatively easy to derive Gibbs sampling algorithms of a new model, which facilitates fast prototype developments.

- With the insights of discrete data studied in this research, the developed Gibbs sampling algorithms are able to tactfully leverage data properties, such as sparsity. Therefore, the developed inference schemes can be relatively efficient for moderately large data.

- With several data augmentation techniques, Gibbs sampling algorithms can be derived for many non-conjugate models, which extends its flexibility and usability.

2.3.3 Characterising Posterior with MCMC Samples

After a Bayesian model is trained with Gibbs sampling (or other MCMC sampling algorithms), we may want to collect samples from the posterior and use the samples to characterise the posterior. The former can be done by using $M_{\text{burnin}} + M_{\text{collection}}$ Gibbs sampling iterations, where we discard the samples in the first M_{burnin} iterations and collect a sample in every J iterations in the later $M_{\text{collection}}$. Collecting samples from multiple independent Markov chains (e.g., multiple runs of a model on the same data with different initialisations and/or random seeds) is also a possible way to collect samples from the posterior.

To analyse the posterior and making predictions, we need to use the samples to characterise the posterior. Consider a model of Eq. (2.1), where we can compute the following properties of the posterior by averaging over S collected samples:

$$\text{Posterior mean}: \sum_{s=1}^{S} \boldsymbol{\theta}^{<s>}/S, \tag{2.53}$$

$$\text{Marginal data distribution}: \mathrm{p}(\mathbf{X} \mid \boldsymbol{\alpha}) = \sum_{s=1}^{S} \mathrm{p}(\mathbf{X} \mid \boldsymbol{\theta}^{<s>})/S, \tag{2.54}$$

$$\text{Predictive distribution}: \mathrm{p}(\boldsymbol{x}^* \mid \mathbf{X}) = \sum_{s=1}^{S} \mathrm{p}(\boldsymbol{x}^* \mid \boldsymbol{\theta}^{<s>})/S, \tag{2.55}$$

where $\boldsymbol{\theta}^{<s>}$ is the point estimate of the model parameter with the s^{th} collected sample.

2.4 Summary

In this chapter, the fundamentals of Bayesian analysis, as well as some advanced techniques like data augmentations have been presented. Specifically, the building blocks of Bayesian models such as the common-used probability distributions and the conjugate priors have been elaborated on. I have also covered how to build a model with those building blocks and the basic framework of Bayesian latent factor models. Finally, how to learn a model with Bayesian inference has been introduced.

Chapter 3

Related Work of Bayesian Latent Factor Models for Discrete Data

This chapter elaborates on the research work of Bayesian Latent Factor Models for discrete data in the areas of text analysis, graph analysis, and multi-label learning, which is highly related to this thesis research. In Section 2.2.5 in the above chapter, I have briefly discussed the general framework and notations of BLFMs. Here I elaborate on how specific BLFMs are built and applied in the above areas.

3.1 BLFM for Text Analysis

3.1.1 Fundamentals

Basic Frameworks

With the bag-of-words assumption, a collection of N documents can be formulated into a discrete matrix $\mathbf{X} \in \mathbb{N}^{V \times N}$, each column of which, $\boldsymbol{x}_i \in \mathbb{N}^V$, is the word occurrences of document i. V is the size of the vocabulary of those documents.

As discussed previously, a typical framework of BLFM with K latent factors for the above data can be presented as follows:

$$\mathbf{X} \sim \mathrm{p}(\mathbf{X} \mid \boldsymbol{\Phi}^\top \boldsymbol{\Theta}), \tag{3.1}$$

where $\boldsymbol{\Phi} \in \mathbb{R}_+^{V \times K}$ and $\boldsymbol{\Theta} \in \mathbb{R}_+^{K \times N}$ are the latent representations.

Note that for a latent factor k, $\boldsymbol{\phi}_k \in \mathbb{R}_+^V$ is a distribution/proportion over the words in the vocabulary, indicating the importance of words. In this way, a latent factor can be viewed as a semantic concept represented by several representative words with the largest weights, which is usually called a "topic." Moreover, $\boldsymbol{\phi}_k$ is usually referred to as the topic-word distribution/proportion. Therefore, in text analysis, BLFMs has a more common name, which is *topic models*. On the other hand, each document i is associated with a distribution/proportion over the topics, $\boldsymbol{\theta}_i \in \mathbb{R}_+^V$, which measures the significance of the topics in this document and can be referred to as the document-topic distribution/proportion. With $\boldsymbol{\theta}_i$, the content of a document can be intuitively demonstrated with the proportions of topics, yielding an important property of topic models, known as *interpretability*.

Two fundamental frameworks of topic models are Latent Dirichlet Allocation (LDA) [Blei et al., 2003] and Poisson Factor Analysis (PFA) [Canny, 2004, Zhou et al., 2012a]. Actually, the latter directly follows the above framework, whose generative process of a document is as follows:

$$\boldsymbol{\theta}_i \sim \text{Gamma}(\alpha_0, 1/\beta_0), \tag{3.2}$$

$$\boldsymbol{\phi}_k \sim \text{Dir}(\gamma_0), \tag{3.3}$$

$$\boldsymbol{x}_i \sim \text{Pois}(\boldsymbol{\Phi}\boldsymbol{\theta}_i). \tag{3.4}$$

Different to PFA, LDA imposes the following prior distributions for θ and ϕ:

$$\boldsymbol{\theta}_i \sim \text{Dir}(\alpha_0), \tag{3.5}$$

$$\boldsymbol{\phi}_k \sim \text{Dir}(\gamma_0). \tag{3.6}$$

Moreover, the standard way of generating a document in LDA is a bit different from PFA. Suppose document i has $x_{\cdot i} = \sum_v^V x_{vi}$ words in total and the j^{th} word of it is denoted as w_i^j. Given $\boldsymbol{\theta}_i$ and $\boldsymbol{\Phi}$, w_i^j is generated as follows:

$$z_i^j \sim \text{Cat}(\boldsymbol{\theta}_i), \tag{3.7}$$

$$w_i^j \sim \text{Cat}(\boldsymbol{\phi}_{z_i^j}), \tag{3.8}$$

where $z_i^j \in \{1, \cdots, K\}$ is the topic assignment of word w_i^j.

In the generative process of LDA, the topic assignment of each word is explicitly generated. If we marginalise out the topic assignments, the generative process can be written as:

$$\boldsymbol{x}_i \sim \text{Multi}\left(x_{\cdot i}, \frac{[\sum_k^K \phi_{1k}\theta_{ki}, \cdots, \sum_k^K \phi_{Vk}\theta_{ki}]}{\sum_v^V \sum_k^K \phi_{vk}\theta_{vi}}\right)^1, \tag{3.9}$$

which is in line with the framework shown in Eq. (3.1).

Another difference between PFA and LDA is that in PFA, the length of a document is treated as a latent variable generated from $x_{\cdot i} \sim \text{Pois}\left(\sum_v^V \sum_k^K \phi_{vk}\theta_{vi}\right)$, while LDA assumes that the length is given as an observed variable. Studied in Canny [2004], this distinction may result in performance differences.

Inference

Learning of a topic model is about the inference of the latent variables, θ and ϕ given the word occurrences of the documents. The commonly-used inference algorithms for topic models can be categorised into the following categories, briefly discussed as follows:

- **MCMC sampling:** MCMC sampling, especially Gibbs sampling, is one of the most popular inference algorithms for topic models. Moreover, by marginalising θ and ϕ out, Gibbs sampling can be done by sampling a word's topic assignment conditioned on the topic assignments for the others, known as the *collapsed Gibbs sampling* algorithm [Griffiths and Steyvers, 2004, Lijoi and Prünster, 2010], which has been demonstrated to have good performance in many BLFMs. Although effective, the vanilla Gibbs sampling for topic modelling is inefficient for large collections of documents. There are extensive studies on scalable inference algorithms of MCMC sampling for topic models. The popular ones are: **1)** leveraging sparsity of documents and the space of latent variables, such as in Yao et al. [2009], Li et al. [2014], Yu et al. [2015]; **2)** using Metropolis-Hastings samplers to reduce sampling complexity, such as in Yuan et al. [2015], Chen et al. [2016] **3)** using Stochastic gradient MCMC (SGMCMC) for the global variables (i.e., topic-word distributions) such as in Patterson and Teh [2013], Ma et al. [2017], Cong et al. [2017].

[1] In the case of LDA, the normalisation term, $\sum_v^V \sum_k^K \phi_{vk}\theta_{vi}$ sums to one, as $\boldsymbol{\theta}_i$ and $\boldsymbol{\phi}_k$ are drawn from Dirichlet.

- **Variational inference:** Mean-field variational inference optimised by coordinate descent was used in the original paper of LDA [Blei et al., 2003]. In recent years, there have been extensive studies of improving variational inference for topic models, from Stochastic Variational Inference (SVI) [Hoffman et al., 2013] that learns the global variables with batches of data, to recently-proposed amortized inference such as in Srivastava and Sutton [2017], Miao et al. [2016] that approximates the posterior by directly encoding a document's word occurrences.

- **Hybrid inference methods:** As an individual inference method has its advantages and disadvantages, it is possible to combine multiple inference methods into a hybrid algorithm. For example, Cong et al. [2017] used Gibbs sampling and SGMCMC for the local and global variables in a topic model, respectively and Zhang et al. [2018] applied amortized inference for local variables and SGMCMC for global variables.

- **Other inference methods:** As the inference of topic models belongs to the general problem of learning graphical models, there are many other maximum likelihood estimate methods other than MCMC sampling and variational inference, such as the Method of Moments (MoM) in Anandkumar et al. [2012], matrix factorisation based methods in Arora et al. [2012], and belief propagation based methods in Zeng et al. [2013].

Evaluations

Accurately and comprehensively measuring the performance of a topic model is still an open problem being studied. Currently, the evaluations of topic models can be done in three general aspects: **1)** modelling accuracy measured by predicting heldout words; **2)** topic quality; **3)** downstream applications such as document classification and clustering. For the last one, the topic distribution of a document can be used as the features that encode the document's semantic information. Classification and clustering are usually done by external algorithms such as support vector machines and K-means on those features. Therefore, the first two aspects are presented.

Predicting heldout words The performance of predicting heldout words can be measured by *perplexity*, which is a common metric of text analysis. Computing perplexity requires estimating the probability of the heldout words given a trained model. Different ways of doing this have been developed, the approaches introduced in Wallach et al. [2009] are commonly-used ones. Concretely, after training a model with the training documents, we randomly select some words as the observed words and use the remaining words as the unobserved words in each testing document, then use the observed words to estimate the predictive probability, and finally compute the perplexity of the unobserved words. Specifically, suppose that the matrix of the testing documents is $\mathbf{X}^* \in \mathbb{N}^{V \times N_{\text{test}}}$, which is split into the observed word matrix $\mathbf{X}^{*\text{o}} \in \mathbb{N}^{V \times N_{\text{test}}}$ and the unobserved word matrix $\mathbf{X}^{*\text{u}} \in \mathbb{N}^{V \times N_{\text{test}}}$, where $\mathbf{X}^* = \mathbf{X}^{*\text{o}} + \mathbf{X}^{*\text{u}}$. After training a topic model, we obtain the global topic-word distributions, $\mathbf{\Phi}$. Conditioned on $\mathbf{\Phi}$, the topic distributions of the test documents, $\mathbf{\Theta}^* \in \mathbb{R}_+^{K \times N_{\text{test}}}$ are estimated from the posterior with $\mathbf{X}^{*\text{o}}$. Finally, the perplexity of $\mathbf{X}^{*\text{u}}$ can be computed as follows:

$$\text{Perplexity} = \exp\left(-\frac{1}{x_{\cdot\cdot}^{*\text{u}}} \sum_{i}^{N_{\text{test}}} \sum_{v}^{V} x_{vi}^{*\text{u}} \log \frac{\sum_k^K \phi_{vk}\theta_{vi}^*}{\sum_v^V \sum_k^K \phi_{vk}\theta_{vi}^*}\right), \tag{3.10}$$

where $x_{\cdot\cdot}^{*\text{u}} = \sum_i^{N_{\text{test}}} \sum_v^V x_{vj}^{*\text{u}}$.

Topic quality In addition to perplexity evaluation of modelling accuracy, another measurement of topic models is topic quality, which can be captured by the coherence of the representative words in a topic. Specifically, for topic k, we usually obtain its representative words by ranking the weights of ϕ_k and the coherence of two words can be measured by the probability that the two words co-occur in a reference corpus. Note that the reference corpus can be either the target corpus or an external large corpus such as Wikipedia. Following the above routine, there are several ways of computing topic coherence. In my research, the Normalised Pointwise Mutual Information (NPMI) Aletras and Stevenson [2013], Lau et al. [2014] is commonly used, which can be calculated for topic k with top T words as follows:

$$\mathrm{NPMI}(k) = \sum_{j=2}^{T} \sum_{i=1}^{j-1} \log \frac{p(w_j, w_i)}{p(w_j)p(w_i)} / - \log p(w_j, w_i), \tag{3.11}$$

where $p(w_i)$ is the probability of word i, and $p(w_i, w_j)$ is the joint probability of words i and j that co-occur together within a sliding window in the reference corpus.

More introduction on other measurements of topic coherence can be found in Röder et al. [2015], the authors of which also released a package for computing those measurements[2].

Among the extensive research of topic modelling, next sections review three lines of works that are highly related to this research including models with meta-data, short-text topic models, and deep topic models.

3.1.2 Topic Models with Meta-Data

Given a corpus, topic models mainly work with the word occurrences of the documents to discover topics. While, in practice, in addition to word occurrences, various kinds of meta-data are associated with documents and words in many corpora. At the document level, labels of documents can be used to guide topic learning so that more meaningful topics can be discovered. Moreover, it is highly likely that documents with common labels discuss similar topics, which could further result in similar topic distributions. For example, if we use authors as labels for scientific papers, the topics of the papers published by the same researcher can be closely related. At the word level, semantic/syntactic features are also accessible. For example, there are features regarding word relationships, such as synonyms obtained from WordNet [Fellbaum, 2012], word co-occurrence patterns obtained from a large corpus, and linked concepts from knowledge graphs. It is preferable that words having a similar meaning but different morphological forms, like “dog” and “puppy”, are assigned to the same topic, even if they may barely co-occur in the modelled corpus. Recently, word embeddings generated by GloVe [Pennington et al., 2014] and word2vec [Mikolov et al., 2013], have attracted a lot of attention in natural language processing and related fields. It has been shown that the word embeddings can capture both the semantic and syntactic features of words so that similar words are close to each other in the embedding space. It seems reasonable to expect that these word embedding will improve topic modelling [Das et al., 2015, Nguyen et al., 2015].

Document Meta-Data

As each document may usually have its specific meta-data, it is natural to use the document-topic distribution $\boldsymbol{\theta}_i$ to incorporate the meta-data of the i^{th} document. In general, there are two ways of doing this: the *supervised way* and the *generative way*. In the former way, document meta-data are directly incorporated into the learning of

[2] `http://palmetto.aksw.org`

document-topic distributions while in the latter, besides generating word occurrences, another generative process is applied to generate document meta-data, where document-topic distributions are usually shared in the two generative processes. Moreover, in the generative way, document meta-data contribute to the learning of document-topic distributions by serving as the evidence together with word occurrences in the posterior. The example models fall into the two ways are presented as follows.

Supervised way To incorporate document meta-data in a supervised way, Supervised LDA (sLDA) [Mcauliffe and Blei, 2008] models document labels by learning a generalised linear model with an appropriate link function and exponential family dispersion function. But the restriction of sLDA is that one document can only have one label. Labelled LDA (LLDA) [Ramage et al., 2009] assumes that each label has one corresponding topic and a document is generated by a mixture of the topics. Although multiple labels are allowed, LLDA requires that the number of topics must equal to the number of labels, i.e., exactly one topic per label. As an extension to LLDA, Partially Labelled LDA (PLLDA) [Ramage et al., 2011] relaxes this requirement by assigning multiple topics to a label. The Dirichlet Multinomial Regression (DMR) model [Mimno and McCallum, 2008] regresses a document's Dirichlet parameters on its meta-data document-topic distribution with the logistic-normal transformation. As full conjugacy does not exist in DMR, a part of the inference has to be done by numerical optimisation. Similarly, in the Hierarchical Dirichlet Scaling Process (HDSP) [Kim and Oh, 2017], conjugacy is broken as well since the topic distributions have to be renormalised. Hu et al. [2016a] introduced a Poisson factorisation model with hierarchical document labels, which may not able to be applied to regular topic models as the topic proportion vectors are also unnormalised. Recently, Card et al. [2018] proposed a variational autoencoder based topic model, which incorporates document meta-data in the encoder[3].

Generative way The generative way is usually used in models that incorporate document meta-data that are in complex forms and non-trivial to incorporate, such as document-document citation graphs and co-author graphs. Moreover, those models usually follow the general idea of matrix co-factorisations, which factorises both the word-document matrix and the graph adjacency matrix, with shared latent representations. For example, Relational Topic Model (RTM) [Chang and Blei, 2009] models an additional document network by generating it according to the latent topics of the document pairs. The models in Zhu et al. [2013] and Lim et al. [2013] incorporate document-document links by generating them according to the document-topic distributions from the Poisson distribution and the Gaussian Process, respectively. Lim and Buntine [2016] integrated the way of modelling document links in Zhu et al. [2013] with topic models with hierarchical Pitman-Yor Process. Moreover, other than using graphs to help analyse texts, another related topic is using texts to enhance graph analysis, such as in Gopalan et al. [2014] and Acharya et al. [2015][4].

In my thesis research, I have proposed an efficient supervised way of regressing document-topic distributions on binary document meta-data in Zhao et al. [2017c, 2018a], detailed in Chapter 5.

Word Meta-Data

Compared with document meta-data, word meta-data are usually incorporated in topic-word distributions. In general, a topic model discovers latent topics from the word oc-

[3] Variational autoencoder based topic models will be presented in Section 3.1.4.

[4] More details of graph analysis are presented in Section 3.2.

currences of a corpus, for example, if two words frequently occur in the same documents, they are expected to be semantically related. Therefore, word meta-data such as word embeddings, are commonly used to enhance the information of word occurrences. So it is not hard to imagine that word meta-data are heavily used in modelling short texts, where the word occurrences are not sufficient enough for a model to learn good topics. Therefore, while presenting models with word meta-data, the models with word meta-data for short texts will be mentioned as well. More details of short-text models will be elaborated on in Section 3.1.3.

Leveraging axillary word features such as word embeddings has been a promising direction in topic modelling. For example, Latent Feature LDA (LFLDA) [Nguyen et al., 2015] integrates word embeddings into LDA by replacing the topic-word Dirichlet multinomial component with a mixture of a Dirichlet multinomial component and a word embedding component. Fu et al. [2016] proposed Word-Topic Mixture (WTM) model that combines the idea of LFLDA and Topical Word Embedding (TWE) model [Liu et al., 2015]. Instead of generating word types (tokens), Gaussian LDA (GLDA) [Das et al., 2015] directly generates word embeddings with the Gaussian distribution. [Xun et al., 2016] introduced an extension combining the idea of GLDA and LFLDA, where a set of indicator variables are sampled to choose either generating the type of a word or generating the embedding of a word. GPUDMM [Li et al., 2016] extends Dirichlet Multinomial Mixture [Yin and Wang, 2014, Jin et al., 2011] with word semantic similarity obtained from embeddings for short texts. As an extension of GPUDMM, GPUPDMM Li et al. [2017] relaxes the constraint of DMM that only one topic is active in a document [Nigam et al., 2000] and it models the active number of topics of a document with the Poisson distribution. Shi et al. [2018] recently proposed a matrix factorisation based topic models for short text, which incorporates word-context semantic correlations. The inference of the model is done by a block coordinate descent algorithm.

In terms of my research on utilising word meta-data, I have developed models that incorporate either binary [Zhao et al., 2017c, 2018a] or real-valued word embeddings [Zhao et al., 2017b, 2018c], as well as using word embeddings to help learn more interpretable topic structures [Zhao et al., 2018c], detailed in Chapter 5 and 6.

Leveraging both Document and Word Meta-Data

To my knowledge, the attempts that jointly leverage document and word meta-data are relatively rare. For example, meta-data can be incorporated by first-order logic in Logit-LDA Andrzejewski et al. [2011] and score functions in SC-LDA Yang et al. [2015]. However, the first-order logic and score functions need to be defined for different kinds of meta-data and the definition can be infeasible for incorporating both document and word meta-data simultaneously.

In my research of Zhao et al. [2017c, 2018a], I have developed a more efficient model to incorporate both document and word meta-data into one joint model and done comprehensive study of how the two kinds of meta-data affect the performance and interpretability of the model, detailed in Chapter 5.

3.1.3 Short-text Topic Models

With the rapid growth of the internet, huge amounts of text data are generated in social networks, online shopping and news websites, etc. Those internet-generated texts are usually short, such as tweets and new headlines, which means that the contextual information (i.e. the word occurrences or the bag-of-words) in one individual document is insufficient, causing degraded performance in topic modelling. Many conventional topic models like LDA discover topics purely based on the contextual information of the target corpus.

Therefore, they can suffer from large performance degradation over short texts. Many research effort has been devoted to tackling the sparsity issue in short texts. In addition to the short-text models with word meta-data in Section 3.1.2, another common strategy is to first aggregate short texts into clusters and then perform standard topic modelling techniques over the clusters, which is also known as "pooling." As the aggregated clusters contain richer contextual information than the original documents, better topics can be learned.

As discussed previously, many internet-generated short texts are associated with document meta-data such as authors, categories, and timestamps. Therefore, one straightforward approach is to aggregate short texts to their meta-data. For example, tweets are naturally labelled by users and hashtags. It is reasonable to assume that tweets published by the same users [Hong and Davison, 2010] or with the same hashtags [Mehrotra et al., 2013] are more likely to talk about similar topics. Another feasible aggregation strategy is to pool short texts into clusters by their contextual information instead of meta-data, which involves two main steps: discover latent topics of short texts and do aggregation according to the topics. For example, the SATM model in Quan et al. [2015] aggregates short texts with a two-phase generative process. The first phase follows the standard LDA to generate a set of regular-sized documents (i.e., clusters), and in the second phase each short text document will be generated from the probability distribution over words associated with the regular-sized document that the short text belongs to. In the PTM model [Zuo et al., 2016a], the two steps are learned in a joint way. In the recently developed MIGA model in Zhao et al. [2019a], document meta-data as well as document context are used to aggregate short texts in a joint generative process.

Besides the aforementioned aggregation-based models, various models have been recently developed for modelling short texts. For example, Yan et al. [2013] proposed Biterm Topic Model (BTM), which directly generates word co-occurrences instead of individual words. Zuo et al. [2016b] developed Word Network Topic Model (WNTM) that learns the distribution over topics for each word instead of the topics for each document, so as to tackle the sparsity and imbalance issues in short texts. Recently, Yang et al. [2018] proposed an approach that jointly models normal documents and the associated short texts (e.g. a new article and its user comments), showing that the contextual information in the normal documents can help the learning of the short texts.

The proposed models in this thesis research including Zhao et al. [2017c, 2018a, 2017b, 2018c,b] improve both modelling performance and topic quality on short-text topic modelling, detailed in Chapter 5 and 6.

3.1.4 Deep Topic Models

The term "deep topic models" here is used to denote those models with deep structures, implemented by either multi-layer Bayesian latent variables or deep neural networks in the generative and/or inference processes, which become increasingly popular in topic modelling research.

Multi-layer Bayesian Topic Models

Standard topic models like LDA assume topics are independent, which is an unnatural assumption in many text corpora. To address this limitation, several advances in topic modelling have started exploring the semantic structures of topics, which is referred to as the *topic structure learning* problem.

Correlated Topic Model (CTM) [Lafferty and Blei, 2006] started the study of capturing pairwise correlations of topics by the covariance matrix of the normal distribution, which serves as the prior of document-topic distributions. The correlations discovered by CTM

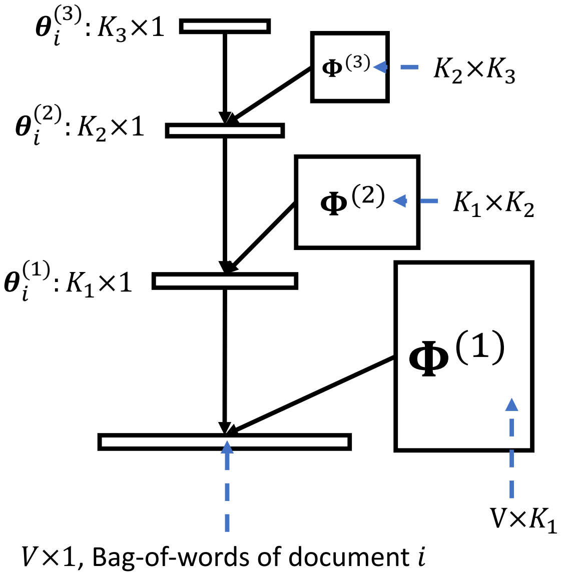

$\boldsymbol{\theta}_i^{(3)}: K_3 \times 1$
$\boldsymbol{\Phi}^{(3)}$
$K_2 \times K_3$
$\boldsymbol{\theta}_i^{(2)}: K_2 \times 1$
$\boldsymbol{\Phi}^{(2)}$
$K_1 \times K_2$
$\boldsymbol{\theta}_i^{(1)}: K_1 \times 1$
$\boldsymbol{\Phi}^{(1)}$
$V \times K_1$
$V \times 1$, Bag-of-words of document i

Figure 3.1: General framework of deep PFA models.

can be viewed as a flat structure of topics. In the same year, Li and McCallum [2006] proposed the Pachinko Allocation to discover topic correlations with a directed acyclic graph (DAG) structure, where the topic nodes occupy the interior levels and the leaves are words. The nested CRP (nCRP) [Blei et al., 2010] and the nested hierarchical DP [Paisley et al., 2015] investigate tree-structured topic hierarchies with the extensions to CRP and HDP, respectively. In a tree-structured topic hierarchy, a topic in a higher layer is supposed to be more general than a topic in a lower layer, which is a natural way of presenting topic structures. Kim et al. [2012b], Ahmed et al. [2013] further extended nCRP by either softening its constraints or applying it to different problems.

Recently, deep extensions of PFA have been proposed such as Gan et al. [2015a], Ranganath et al. [2015], Zhou et al. [2015], Henao et al. [2015], Zhou et al. [2016] to improve both modelling accuracy and interpretability. Shown in Figure 3.1, those models follow a common framework, which applies hierarchical factorisations to the topic distribution of a document. Specifically, in the bottom layer, a PFA framework is applied to generate the word occurrences of document i, where $\boldsymbol{\theta}_i^{(1)}$ and $\boldsymbol{\Phi}^{(1)}$ are the first-layer document-topic distribution and topic-word distributions, respectively. Next, $\boldsymbol{\theta}_i^{(1)}$ is further factorised with $\boldsymbol{\theta}_i^{(2)}$ and $\boldsymbol{\Phi}^{(2)}$, where whether $\boldsymbol{\Phi}^{(1)}$ can be interpreted depends on specific model configurations. From the neural network point of view, $\boldsymbol{\theta}$ is the latent representation in each layer and $\boldsymbol{\Phi}$ is the connection weights between two adjacent layers. With this framework, different models apply different prior constructions of each layer's latent representations and connection weights. For example, DPFA [Gan et al., 2015b] uses a binary latent vector for each document in each layer connected by Gaussian weights, DPFM [Henao et al., 2015] assumes each document in each layer has a gamma vector with a binary vector as its focusing indicator, which is drawn from the Bernoulli-Poisson link[5], GBN [Zhou et al., 2015, 2016] uses a vector of gamma variables connected by the weights drawn from Dirichlet distributions, and DEF [Ranganath et al., 2015] is a general framework of deep structure on $\boldsymbol{\theta}_i$, where different constructions of latent representations and connection weights are used.

With careful designs of the latent representations and the connection weights, besides improving modelling performance, above deep models are able to discover interpretable

[5] Detailed in Eq. (3.52).

topic structures. For example, in the GBN model [Zhou et al., 2015, 2016], $\mathbf{\Phi}^{(2)} \in \mathbb{R}_{+}^{K^1 \times K^2}$ models the weights between the second-layer and the first-layer topics. Therefore, the second-layer topics can be interpreted by $\mathbf{\Phi}^{(1)}\mathbf{\Phi}^{(2)}$.

Although with nice properties, deep structures of those models may complicate the inference procedures, leading to less scalable performance on large datasets. Therefore, scalable inference algorithms have been developed for those models, including SGMCMC in Gan et al. [2015a], Cong et al. [2017], blackbox variational inference in Ranganath et al. [2015], and amortized variational inference in Zhang et al. [2018].

Different from the above models, my work in Zhao et al. [2018b] studies a deep factorisation model on the topic-word distributions instead of the document-topic distributions, which has several appealing properties over the previous ones, detailed in Chapter 6.

Neural Topic Models

Recently, deep generative models such as Variational Autoencoders (VAEs) [Rezende et al., 2014] have become increasingly popular for modelling real-valued data, such as images. The success of VAEs has motivated machine learning practitioners to adapt VAEs to dealing with discrete data as done in recent works [Miao et al., 2016, 2017, Krishnan et al., 2018, Liang et al., 2018]. Instead of using the Gaussian distribution as the data distribution for real-valued data, the multinomial distribution has been used for discrete data [Miao et al., 2016, Krishnan et al., 2018, Liang et al., 2018]. Following Liang et al. [2018], we refer to these VAE-based models as "MultiVAE" (Multi for multinomial)[6]. MultiVAE can be viewed as a deep nonlinear PMF model, where the nonlinearity is introduced by a deep neural network in the decoder. Compared with conventional hierarchical Bayesian models, MultiVAE increases its modelling capacity without sacrificing the scalability, because of the use of amortized variational inference (AVI) [Rezende et al., 2014]. This makes MultiVAE a good choice for large-scale discrete data.

Compared to the extensive applications in image analysis, VAEs for discrete data are relatively rare. Miao et al. [2016] proposed the Neural Variational Document Model (NVDM), which extended the standard VAE with a multinomial likelihood for document modelling and Miao et al. [2017] further built a VAE to generate the document-topic distributions in the LDA framework. Srivastava and Sutton [2017] developed an AVI algorithm for the inference of LDA, which can be viewed as a VAE model, although the generative process is the same as the original LDA. Card et al. [2018] introduced a general VAE framework for topic modelling, which is able to incorporate meta-data. Krishnan et al. [2018] recently found that using the standard training algorithm of VAEs in large sparse discrete data may suffer from model underfitting and they proposed Nonlinear Factor Analysis (NFA) with a stochastic variational inference (SVI) [Hoffman et al., 2013] algorithm initialised by AVI to mitigate this issue. In the collaborative filtering domain, Liang et al. [2018] noticed a similar issue and alleviated it by proposing MultiVAE with a training scheme based on KL annealing [Bowman et al., 2016]. Note that the generative processes of NVDM, NFA, and MultiVAE are very similar but their inference procedures are different. NFA is reported to outperform NVDM on text analysis [Krishnan et al., 2018] while MultiVAE is reported to have better performance than NFA on collaborative filtering tasks [Liang et al., 2018]. Recently, instead of using multinomial likelihood, Zhao et al. [2019b] proposed a VAE model based on the negative binomial likelihood, which captures overdispersion in count-valued data.

Although the above neural models have shown great potential on topic modelling, the deep structures in them are usually not able to be interpreted like in the hierarchical

[6] In terms of the generative process (encoder), the models in Miao et al. [2016], Krishnan et al. [2018], Liang et al. [2018] are similar, despite that the inference procedures are different.

Table 3.1: Notations of SBM.

N	the number of nodes
K	the number of latent communities
$z \in \mathbb{N}_+^N$	the latent community indexes of all nodes
$z_i \in \{1, \ldots, K\}$	the latent community index of node i
$\mathbf{M} \in \mathbb{R}^{K \times K}$	the stochastic block matrix
$\pi \in \mathbb{R}_+^K$	the probability vector over latent communities
$\pi_i \in \mathbb{R}_+^K$	the probability vector over latent communities of node i

Bayesian models presented previously. Improving the interpretability of those neural topic models is still an open yet important research direction.

3.2 BLFM for Graph Analysis

Graph analysis has been an extremely important research area in recent years. Among the extensive methods in this area, here I will focus on stochastic random graph models [Miller et al., 2009, Zhou, 2015, Caron and Fox, 2017] based on the Bayesian version of Stochastic Block Models (SBMs), which have been successfully used in relational graph analysis on the tasks of community detection and link prediction.

3.2.1 Fundamentals

SBM is a fundamental model for relational graph analysis, firstly introduced in Wang and Wong [1987] and further developed in Nowicki and Snijders [2001]. In the original SBMs, there are a finite number of latent communities in a graph, each node of which belongs to one latent community. The structure of the graph is assumed to be determined by the latent communities. The basic notations of SBMs are shown in Table 3.1, where $m_{k_1k_2}$ gives the connection strength between latent communities k_1 and k_2. Given z and $\mathbf{M}$, we can write down the data likelihood of the adjacency matrix of a graph $\mathbf{X}$ with N nodes as:

$$\begin{aligned} \mathrm{p}(\mathbf{X} \mid z, \mathbf{M}) &= \prod_{i=1,j=1}^{N,N} \mathrm{p}(X_{ij} \mid z_i, z_j, \mathbf{M}), && (3.12) \\ x_{ij} &\sim \mathrm{Bern}\left(\sigma(m_{z_iz_j})\right), && (3.13) \end{aligned}$$

where $\sigma(\cdot)$ is a function transforming values on $(-\infty, \infty)$ to $(0, 1)$, a common choice of which is the logistic function $\sigma(x) = \frac{1}{1+e^{-x}}$.

Stochastic Block Models

In the Bayesian version of SBMs, one can impose prior distributions of the latent variables as follows:

$$\begin{aligned} \pi &\sim \mathrm{Dir}(\alpha), && (3.14) \\ z_i &\sim \mathrm{Cat}(\pi), && (3.15) \\ m_{k_1k_2} &\sim \mathrm{Beta}(a, b), && (3.16) \\ x_{ij} &\sim \mathrm{Bern}(m_{z_iz_j}). && (3.17) \end{aligned}$$

Note that SBMs can be applied to both directed ($\mathbf{M}$ is asymmetric) and undirected graphs ($\mathbf{M}$ is symmetric). Moreover, SBMs are able to generate assortative/disassortative graphs by setting $\mathbf{M}$ diagonal/off-diagonal dominant. The word "assortative" describes a type of network structure where entities tend to connect to entities belonging to the same latent communities (having similar attributes) while "disassortative" describes the reverse trend. Assortative structures are commonly observed in relational graphs, for example in social graphs, people are likely to connect others within the same community and so are disassortative structures.

As a Bayesian SBM can be viewed as a specific kind of BLFMs, the inference is similar to the ones introduced in topic models, detailed in Section 3.1.1.

Link Prediction Evaluation

A straightforward idea of evaluating a graph analysis model is to check its performance on predicting missing links, where some commonly-used measurements are presented as follows.

AUC-ROC and AUC-PR AUC-ROC and AUC-PR are abbreviations of *area under receiver operating characteristic curve* and *area under precision recall curve*, respectively. Given the probability of a missing link between node i and j predicted by a model, $\mathrm{p}(x_{ij} = 1)$, one can get different predictions, by varying a threshold c, as follows:

$$\begin{cases} x_{ij} = 1, & \text{if } \mathrm{p}(x_{ij} = 1) > c \\ x_{ij} = 0. & \text{otherwise} \end{cases} \tag{3.18}$$

Given the ground-truth links in the test set, AUC-ROC is created by plotting the true positive rate (TPR or recall) against the false positive rate (FPR) at various threshold settings. Similarly, AUC-PR can also be plotted. The TPR, FPR, and precision can be computed as:

$$\text{TPR/recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}}, \tag{3.19}$$

$$\text{FPR} = \frac{\text{false positives}}{\text{false positives} + \text{true negatives}}, \tag{3.20}$$

$$\text{precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}}. \tag{3.21}$$

Higher AUC-ROC or AUC-PR means a model has a better capacity of distinguishing positive (the existence of a link) and negative (non-existence of a link). According to Davis and Goadrich [2006], if a curve dominates in precision recall space, it also dominates in ROC space but the reverse is not guaranteed. While in practice, both measures are usually reported.

Mean rank and hits@10 For large-scale graphs, computing AUC-ROC or AUC-PR is intractable, because we need to compute the statistics by iterating over different thresholds. Proposed in Bordes et al. [2013], mean rank and hits@10 are measurements for information retrieval performance. Given a test node i, one first ranks the all the possible node pairs of $\{(i, j)\}_{j=1,\cdots,N}$ by $\mathrm{p}(x_{i,j} = 1)$. After that, mean rank can be computed by the average rank of true node pair in the above ranks, while hits@10 calculates wither the true node pair is ranked within top 10 in the above rank. A lower mean rank is better while a higher Hit@10 is better.

Mean square error and mean absolute error For weighted networks (count weights or continuous weights), mean square error (MSE) and mean absolute error (MAE) can be used to measure the closeness of ground-truth weights and predicted weights.

In addition to the above measurements, log likelihood and perplexity[7], which are commonly used for evaluating probabilistic models, can also be used in the evaluation of BLFMs for graph analysis.

3.2.2 BLFMs in the SBM Framework for Graph Analysis

Here the representative BLFMs in line with the framework of SBM for graph analysis are reviewed. The directions of these models can be summarised as follows:

- **Automatically selecting K:** The number of latent communities, K, is a parameter that needs to be set in the original SBMs. The time complexity deteriorates to $O(N^5)$ when doing the model selection with K [Yang and Zhao, 2015]. A popular solution is to use Bayesian non-parametric techniques automatically choosing K that fits the data.
- **Allowing each node to belong to multiple latent communities:** A constraint of the original SBMs is one node can only belong to one latent community, which could be an unnatural assumption for many graphs. Various models have been proposed to relax this constraint.
- **Modelling complex structures of latent communities:** The original SBMs usually impose a flat structure on latent communities, which restricts its ability to model complex factors of relational graphs. Models of discovering hierarchical structures of latent communities have recently gained popularity.
- **Scaling to large-scale graphs:** The original SBMs scale quadratically in the number of nodes, which is intractable with large-scale graphs. Different approaches have been developed to improve the efficiency of SBMs.

Now the details of some example models are as follows.

Infinite relational model (IRM) [Kemp et al., 2006] relaxes the first constraint of the original SBM by replacing the parametric Dirichlet distribution with a non-parametric Griffiths-Engen-McCloskey (GEM) [Pitman et al., 2002] distribution as follows:

$$\begin{aligned} \pi &\sim \mathrm{GEM}(\alpha) && (3.22)\\ z_i &\sim \mathrm{Cat}(\pi), && (3.23)\\ m_{k_1k_2} &\sim \mathrm{Beta}(a, b), && (3.24)\\ x_{ij} &\sim \mathrm{Bern}(m_{z_iz_j}). && (3.25) \end{aligned}$$

Inference of IRM is done by Gibbs sampling in the original paper. Shown in Palla et al. [2012], the time complexity of computing the likelihood of IRM is $O(K^2L)$ where L is the number of links in the network. This allows excellent scalability of IRM on typical sparse real-world networks where the number of links is much smaller than the number of non-links.

Bayesian community detection (BCD) [Mørup and Schmidt, 2012] extends IRM by making the stochastic block matrix diagonal dominant:

$$\begin{aligned} \pi &\sim \mathrm{GEM}(\alpha), && (3.26)\\ z_i &\sim \mathrm{Cat}(\pi), && (3.27)\\ \gamma_k &\sim \mathrm{Beta}(b, b), && (3.28)\\ m_{k_1k_2} &= \begin{cases} \mathrm{Beta}(a, a), & \text{if } k_1 = k_2 = k \\ \mathrm{BetaInc}(a, a, \lambda_{k_1k_2}), & \text{otherwise} \end{cases} && (3.29) \end{aligned}$$

[7] Details in Section 3.1.1.

where $\lambda_{k_1k_2} = \min(\gamma_{k_1}M_{k_1k_1}, \gamma_{k_2}M_{k_2,k_2})$ and $\text{BetaInc}(a, b, \lambda)$ is the incomplete beta distribution constrained to the interval $(0, \lambda)$.

Infinite mixed membership model (IMMM) [Koutsourelakis and Eliassi-Rad, 2008] extends IRM with a hierarchical Dirichlet process:

$$\pi_0 \sim \text{GEM}(\alpha_0) \tag{3.30}$$

$$\pi_i \sim \text{DP}(\alpha, \pi_0) \tag{3.31}$$

This extension allows IMMM to assign a node to multiple latent communities and introduces more sharing among the latent communities.

Mixed membership stochastic block model (MMSB) [Airoldi et al., 2009] aims to tackle the second constraint of SBMs, by allowing nodes to have mixed memberships:

$$\begin{aligned} \pi_i &\sim \text{Dir}(\alpha), && (3.32)\\ m_{k_1k_2} &\sim \text{Beta}(a, b), && (3.33)\\ &\text{for each pair } (i, j) \text{ and } i, j = 1, \ldots, N \\ z_i^{(ij)} &\sim \text{Cat}(\pi_i), && (3.34)\\ z_j^{(ij)} &\sim \text{Cat}(\pi_j), && (3.35)\\ x_{ij} &\sim \text{Bern}(m_{z_i^{(ij)}, z_j^{(ij)}}), && (3.36) \end{aligned}$$

where $\pi, \alpha \in \mathbb{R}_+^K$.

The difference between MMSB and the original SBMs, is that the former assumes each node has its own probability vector over latent communities while the latter assumes that nodes share the same probability vector. This difference makes MMSB able to "softly" assign a node to multiple latent communities. A nested variational EM is proposed for learning the model. The proposed algorithm is claimed to outperform Gibbs samplers in terms of memory requirements and convergence rates. Note that in the original MMSB, the stochastic block matrix is a parameter of the model as well as the Dirichlet concentration α. Both of them are estimated by a variational EM algorithm. A variation of MMSB proposed in the original paper is modelling sparsity by down-weighting the probability of the existence of a link to $(1 - \rho) \cdot M_{z_i, z_j}$ where $\rho \in (0, 1)$.

Assortative mixed membership stochastic block model (aMMSB) [Gopalan et al., 2012] is a subclass of MMSB for community detection with the assumption of assortativity by setting the stochastic block matrix diagonal dominant:

$$\beta_k \sim \text{Beta}(a, b), \tag{3.37}$$

$$m_{k_1k_2} = \begin{cases} \beta_k, & \text{if } k_1 = k_2 = k \\ \epsilon. & \text{otherwise} \end{cases} \tag{3.38}$$

Assortative MMSB with node popularities (AMP) [Gopalan et al., 2013] extends aMMSB with introducing node popularities:

$$\theta_i \sim \mathcal{N}(0, \sigma_\theta^2), \tag{3.39}$$

$$\beta_k \sim \mathcal{N}(0, \sigma_\beta^2), \tag{3.40}$$

$$x_{ij} \sim \text{Bern}\left(\sigma(\theta_i + \theta_j + m_{z_iz_j})\right), \tag{3.41}$$

where θ_i captures the popularity of node i and $\sigma(x) = \frac{1}{1+e^{-x}}$ is the logistic function.

The intuition behind AMP is in addition to assortativity, the probability of the existence of a link between node i and j is determined by their popularities as well, meaning that two nodes are likely to get connected if they are popular [Papadopoulos et al., 2012]. Taking popularity into account makes AMP outperform aMMSB significantly.

Hierarchical Dirichlet Process Relational Model (HDPR) [Kim et al., 2013] extends aMMSB with a hierarchical Dirichlet process (similar to IMMM's extension to IRM):

$$\pi_0 \sim \mathrm{GEM}(\gamma), \tag{3.42}$$

$$\pi_i \sim \mathrm{DP}(\alpha, \pi_0). \tag{3.43}$$

Non-parametric latent feature model (NLFM) [Miller et al., 2009] addresses the first two constraints of SBM by the Indian Buffet Process (IBP) [Ghahramani and Griffiths, 2006]:

$$\mathbf{Z} \sim \mathrm{IBP}(\alpha), \tag{3.44}$$

$$m_{k_1k_2} \sim \mathcal{N}(0, \sigma_m^2), \tag{3.45}$$

$$x_{ij} \sim \mathrm{Bern}\left(\sigma(z_i^\top \mathbf{M} z_j)\right), \tag{3.46}$$

where $\mathbf{Z} \in \{0,1\}^{K\times N}$, indicating the which communities a node is assigned to.

Compared with MMSB, NLFM applies a different way of solving the second constraint of SBM. For example, in a social network, a model may discover two latent communities: athletes and musicians. If a person belongs to the two latent communities at the same time, it means the person is an athlete and musician in NLFM while an MMSB model would say the more the person is an athlete, the less he is a musician. Moreover, the stochastic block matrix in NLFM is drawn from Gaussian distribution, so negative values are allowed.

As a simple extension is also proposed in the original paper to incorporate the attributes (meta-data) of entities or links:

$$x_{ij} \quad \sim \quad \mathrm{Bernoulli}\left(\sigma(z_i^\top \mathbf{M} z_j + \beta y_{ij})\right), \tag{3.47}$$

where y_{ij} represents the attribute of the link between i and j and β is the weight associated with the attribute.

The time complexity of computing the likelihood of NLFM is $O(K^2N^2)$. Inference of Z and M is done by Gibbs sampling and Metropolis-Hastings sampling respectively.

Infinite multiple membership relational model (IMRM) [Mørup et al., 2011] scales up NLFM from $O(K^2N^2)$ to $O(K^2L)$. Recall in NLFM, the probability of the existence of a link is modelled as:

$$\mathrm{p}(x_{ij} = 1) = \frac{1}{1 + e^{-z_i^\top \mathbf{M} z_j}}. \tag{3.48}$$

Instead of using the sigmoid function above, IMRM uses a noisy-or function:

$$\mathrm{p}(x_{ij} = 1) = 1 - \prod_{k_1,k_2} (1 - m_{k_1k_2})^{z_{k_1i}z_{k_2j}} = 1 - e^{z_i^\top \log(1-\mathbf{M}) z_j}. \tag{3.49}$$

Then we can rewrite the likelihood as:

$$\mathrm{p}(\mathbf{X}) = \prod_{(i,j)\in\mathcal{X}_1} \left(1 - e^{z_i^\top \log(1-\mathbf{M}) z_j}\right)^{x_{ij}} \prod_{(i,j)\in\mathcal{X}_0} e^{z_i^\top \log(1-\mathbf{M}) z_j}, \tag{3.50}$$

where $\mathcal{X}_1$ and $\mathcal{X}_0$ are the sets of links and non-links in the observed data, respectively.

The exponent of the second term can be efficiently computed as:

$$\sum_{k_1,k_2} \log(1 - m_{k_1k_2}) \left(\sum_{i=1}^{N} z_{k_1i} \sum_{j=1}^{N} z_{k_2j} - \sum_{(i,j)\in\mathcal{X}_1\cup\mathcal{X}_?} z_{k_1i}z_{k_2j} \right), \tag{3.51}$$

where $\mathcal{X}_?$ is the set of unknown links.

If a graph is dominated by non-links, i.e, the graph is sparse, the computation of the likelihood scales linearly in the number of network links. This property makes IMRM able to apply in large relational networks. A similar idea is proposed in Zhou [2015] where the Poisson-Bernoulli link is used to achieve the same goal.

Inference of Z and M is done by the split-merge sampling of IBP and Hamiltonian Markov chain Monte Carlo (HMC) [Neal et al., 2011] respectively.

Infinite edge partition model (EPM) [Zhou, 2015] assumes each link in a relational network is generated by a Bernoulli-Poisson link:

$$\begin{aligned} x_{ij} &= \mathrm{I}(s_{ij} \geq 1), & (3.52)\\ s_{ij} &\sim \mathrm{Pois}(\lambda_{ij}), & (3.53) \end{aligned}$$

meaning that two nodes are connected if they interact at least once.

Using the Bernoulli-Poisson link gives us two advantages: **1)** The binary-modelling problem is transformed into a count-modelling one, which makes it easier to build hierarchical Bayesian models. **2)** Similar to IMRM, it has a property that if $x_{ij} = 0$ then $s = 0$. It means we only need to handle the cases of $x_{ij} = 1$ and this leads to significant computational savings.

if s is marginalised out, we get:

$$x_{ij} \mid \lambda_{ij} \sim \mathrm{Bernoulli}(1 - e^{-\lambda_{ij}}). \tag{3.54}$$

The generative process of EPM is:

$$\begin{aligned} \pi_i &\sim \mathrm{Gamma}(a_i, 1/b_i), & (3.55)\\ r_k &\sim \mathrm{Gamma}(\gamma_0/K, 1/c_0), & (3.56)\\ m_{k_1k_2} &= \begin{cases} \mathrm{Gamma}(\epsilon r_k, d), & \text{if } k_1 = k_2 = k \\ \mathrm{Gamma}(r_{k_1} r_{k_2}, d), & \text{otherwise} \end{cases} & (3.57)\\ x_{ij} &\sim \mathrm{Bernoulli}(1 - e^{-\pi_i^\top \mathbf{M} \pi_j}), & (3.58) \end{aligned}$$

where $a_i, b_i, \gamma_0, c_0, \epsilon$ are drawn from gamma distributions.

Described in Section 2.2.4, the construction of $\boldsymbol{r}$ is actually a truncated gamma process. Therefore, the construction of $\mathbf{M}$ can be viewed as an extension of a gamma process, which automatically learns the number of communities, named hierarchical relational gamma process in the paper

Infinite latent attribute model (ILA) [Palla et al., 2012, 2015] extends NLFM with a two-layer hierarchy on latent communities:

$$\begin{aligned} \mathbf{Z} &\sim \mathrm{IBP}(\alpha), & (3.59)\\ \pi_k &\sim \mathrm{GEM}(\gamma), & (3.60)\\ z'_{ik} &\sim \mathrm{Cat}(\pi_k), & (3.61)\\ m_{kk'_1k'_2} &\sim \mathcal{N}(0, \sigma_m^2), & (3.62)\\ x_{ij} &\sim \mathrm{Bernoulli}\left(\sigma(\sum_{k=1}^{\infty} m_{kz'_{ki}z'_{kj}})\right), \text{for } z_{ki} = 1, \text{ and } z_{kj} = 1 & (3.63) \end{aligned}$$

In ILA, an entity first chooses a set of latent communities (drawn from IBP) and then chooses one sub-community (drawn from GEM and the categorical distribution) for each latent community it belongs to. The time complexity of IRA is $O(KN^2)$. Inference of $\mathbf{Z}$ and $\mathbf{Z}'$ is done by Gibbs sampling and inference of $\mathbf{M}$ is done by Metropolis-Hastings sampling or slice sampling.

Models with tree-structured hierarchy Going beyond the two-layer hierarchies, it is possible to impose tree-structured hierarchies on the latent communities, where a community can contain sub-communities or be contained by super-communities. The basic idea of models introduced here is the probability of a link between i and j is determined by their common ancestor(s) in the tree. Following this basic procedure, various models are proposed with different tree constructions. Hierarchical random graphs (HRG) model [Clauset et al., 2008] applies a uniform binary tree on the latent class structure. Infinite tree-structured model (ITSM) [Herlau et al., 2012] extends HRG by replacing the binary tree with a uniform multifurcating tree and also drawing the first level latent class from a CRP. The Mondrian process (MP) is a non-parametric prior distribution over kd-tree structures. The generative process can be viewed as splicing up a rectangle into blocks. Roy and Teh [2009] used MP to construct the tree structure that divides entities into different blocks. Multi-scale community block model (MCSB) [Ho et al., 2011] used the nested CRP to build the tree structure which releases the limit of binary hierarchies in MP. Each entity is allowed to belong to the latent communities of a path with different probabilities, which captures the multi-scale granularity of the hierarchies.

Relationships of above models: Figure 3.2 shows the relationships of the above models.

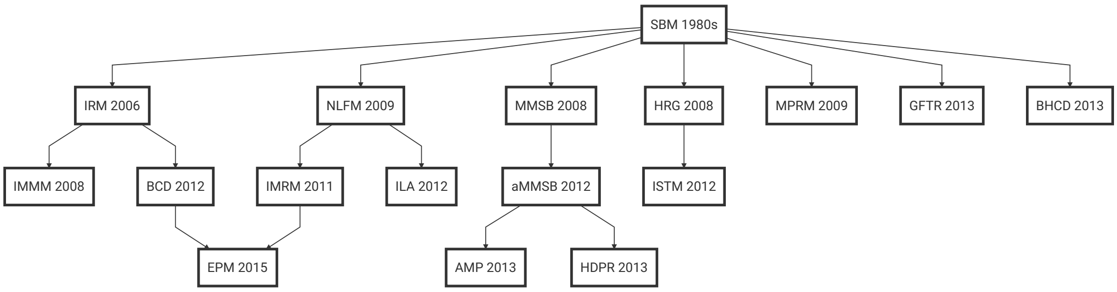


Figure 3.2: Relations of the discussed BLFMs in the SBM framework. The arrows indicate the extension relationship.

3.2.3 BLFMs with Meta-Data for Graph Analysis

Now BLFMs with meta-data in the area of graph analysis are reviewed. Note that meta-data associated with a graph can be either on the links or on the nodes. Here I am particularly interested in the latter case.

Similar to topic models that incorporate meta-data discussed in Section 3.1.2, the general idea of incorporating a node's meta-data in BLFMs is to regress the distribution over latent communities of the node, i.e., $\boldsymbol{\pi}_i$ on the meta-data. Following this idea, different ways of constructing the regression have been proposed.

Suppose that each node i in a graph is associated with some meta-data encoded in $\boldsymbol{y}_i$, which is a L dimensional binary/count-valued/real-valued vector. As denoted before, $\boldsymbol{\pi}_i$ is a K dimensional vector over the communities for node i. Nonparametric Meta-data Dependent Relational Model (NMDR) [Kim et al., 2012a] uses the stick-breaking construction of a DP to incorporate node meta-data as follows:

$$\boldsymbol{\nu}_i \sim \mathcal{N}(\mathbf{W}\boldsymbol{y}_i, -)^8 \tag{3.64}$$

$$\pi_{ki} = \sigma(\nu_{ki}) \prod_{k'}^{k-1} \sigma(-\nu_{k'i}), \tag{3.65}$$

[8] Here "$-$" is used to denote the parameter that is out of interest.

where $\sigma(x) = \frac{1}{1+e^{-x}}$ is the logistic function and $\mathbf{W} \in \mathbb{R}^{K \times L}$ is the regression weight matrix drawn from the normal distributions.

Hierarchical Dirichlet scaling process (HDSP) [Kim et al., 2012a] generates the probability vector constructed by normalising a vector of gamma random variables, whose scale parameters are used to incorporate meta-data:

$$\phi_i \sim \text{Gamma}\left(-, \frac{1}{\sigma(\mathbf{y}_i)}\right), \tag{3.66}$$

$$\pi_i = \phi_i / \phi_{\cdot i}, \tag{3.67}$$

where different options of $\sigma(x)$ are proposed.

Proposed in Fan et al. [2017], Node-Information Involved Mixed-Membership Model (niMM) uses the stick-breaking constructions of DP extending IMMM [Koutsourelakis and Eliassi-Rad, 2008] with binary node meta-data, as follows:

$$\psi_{ki} \sim \text{Beta}\left(1, \prod_{l=1}^{L} w_{kl}^{y_{li}}\right), \tag{3.68}$$

$$\pi_{ki} \sim \psi_{ki} \prod_{k'=1}^{k-1} (1 - \psi_{k'i}), \tag{3.69}$$

where w_{kl} is the weight connecting community k and meta-data l, drawn from the gamma distribution.

In addition, Node-Information Involved Latent Feature Model (niLF) extends NLFM [Miller et al., 2009] with binary node meta-data by the stick-breaking construction of IBP, as follows:

$$\psi_{ki} \sim \text{Beta}\left(\prod_{l=1}^{L} w_{kl}^{y_{li}}, 1\right), \tag{3.70}$$

$$\pi_{ki} \sim \prod_{k'=1}^{k} \psi_{k'i}, \tag{3.71}$$

$$z_{ki} \sim \text{Bern}(\pi_{ki}). \tag{3.72}$$

In Hu et al. [2016a], a summation model is proposed to incorporate binary meta-data, which generates an unnormalised gamma vector:

$$\pi_{ki} \sim \text{Gamma}\left(\sum_{l:y_{li}>0} w_{kl}, -\right), \tag{3.73}$$

where w_{kl} is the weight connecting community k and meta-data l, drawn from the gamma distribution. This construction can be extended with hierarchical meta-data.

In Zhao et al. [2017a], I have developed a novel way of incorporating binary node meta-data, which has several advantages over the above ones, detailed in Chapter 4.

3.3 BLFM for Multi-label Learning

3.3.1 Fundamentals

Multi-label learning is a classification problem with unique properties, such as the label space can be very large and sparse and there can be a large proportion of missing labels for some samples, which make conventional classification approaches infeasible. Suppose $\mathbf{X} \in \mathbb{R}^{F \times N}$ is the feature matrix for N data samples, where F is the number of the feature dimensions. For each sample i, there is a vector of labels $\boldsymbol{y}_i \in \{0,1\}^L$, where L is the number of the label dimensions. $\mathbf{Y} \in \{0,1\}^{L \times N}$ is referred to as the label matrix. The task of multi-label learning is to predict the labels of a new sample after a model is trained with the above training data.

An important line of models for multi-label learning is called the label embedding methods, which project the high-dimensional sparse label vectors of each instance into a low-dimensional space. However, learning the embedding itself is a computationally challenging problem, especially when the label matrix is massive. This has led to a lot of recent interest in embedding based models for multi-label learning that can learn label matrix embeddings efficiently [Yu et al., 2014, Mineiro and Karampatziakis, 2015]. On the other hand, it is known that BLFMs, which can be viewed as the Bayesian version of embedding methods, have advantages of dealing with sparsity and missing data, making them potential candidates for multi-label learning. A general idea is that each sample is associated with a set of low-dimensional latent variables (embeddings), which generate both the label vector and feature vector or generate the label vector conditioned on the feature vector. As the label matrix is discrete, if we treat the feature matrix as the meta-data, the techniques of the topic models and graph models with meta-data presented previously, are able to be applied to solve the multi-label problem.

Given the size of the label matrix, the ranking based measurements of information retrieval are commonly used as the evaluations of multi-label learning problems, such as Recall@R and the truncated normalized discounted cumulative gain (NDCG@R). The two measurements for a testing sample i with the label vector $\boldsymbol{y}^*$ can be computed as follows[9]:

$$\text{Recall@}R = \frac{\sum_{r=1}^{R} \mathrm{I}\left(y^*_{\omega(r)i} = 1\right)}{\min(R, y^*_{:i})}, \tag{3.74}$$

$$\text{DCG@}R = \frac{\sum_{r=1}^{R} 2^{\mathrm{I}\left(y^*_{\omega(r)i}=1\right)} - 1}{\log(r+1)}, \tag{3.75}$$

where $\omega(r) \in \{1, \cdots, L\}$ is the label at rank r, which is obtained by sorting the predictive probabilities of the labels of a user; $\mathrm{I}\left(y^{*u}_{\omega(r)i} = 1\right)$ indicates whether the label is actually active for sample i; NDCG@R is computed by linearly normalising DCG@R into $[0,1]$. Intuitively, Recall@R measures the number of the R predicted items that are within the set of the ground-truth items but does not consider the item rank in R, while NDCG@R assigns larger discounts to lower ranked items.

Next, I will stick on reviewing BLFMs applied in the multi-label learning problem and present several representative works in this line.

3.3.2 BLFMs for Multi-Label Learning

Kapoor et al. [2012] first introduced a Bayesian model for multi-label learning approximating a sample's latent vector via the transformations of its features, which serves as the means of a normal distribution generating the sample's labels. This model can be viewed as a BLFM, whose connections of the latent variables are defined by parametric functions. Bi and Kwok [2014] introduced a model which regresses a sample's latent vector on its feature vector, which generates its labels from the normal distribution, detailed as follows:

$$\boldsymbol{\theta}_i \sim \mathcal{N}\left(\mathbf{W}^\top \boldsymbol{x}_i, \boldsymbol{\Omega}\right), \tag{3.76}$$

$$\boldsymbol{y}_i \sim \mathcal{N}\left(\boldsymbol{\theta}_i, -\right), \tag{3.77}$$

[9] More metrics can be found at `http://manikvarma.org/downloads/XC/XMLRepository.html`.

where $\mathbf{W} \in \mathbb{R}^{F \times L}$ is the weights that regress the labels of a sample on its features, drawn from the normal distribution and the covariance matrix $\boldsymbol{\Omega}$ can be used to encode the label correlations.

One representative example of using the techniques of LDA, PFA and related models is proposed by Rai et al. [2015], which applies the following process to generate the labels of sample i with the Bernoulli-Poisson link conditioned on its features:

$$\begin{aligned}
\phi_k &\sim \text{Dir}(-), && (3.78)\\
p_{ki} &= \sigma(\boldsymbol{w}_k^\top \boldsymbol{x}_i), && (3.79)\\
\theta_{ki} &\sim \text{Gamma}\left(-, \frac{p_{ki}}{1-p_{ki}}\right), && (3.80)\\
m_{li} &\sim \text{Pois}(\boldsymbol{\Phi}\boldsymbol{\theta}_i), && (3.81)\\
y_{li} &\sim \text{I}(m_{li} > 0), && (3.82)
\end{aligned}$$

where $\sigma(a) = \frac{1}{1+e^{-a}}$ is the logistic function and $\mathbf{W} \in \mathbb{R}^{F \times K}$ is the weight matrix that regresses the latent representations of a sample on its features, drawn from the normal distributions. As the Bernoulli-Poisson link is a good choice for modelling sparse data, it makes this model be able to capture the sparsity in the label matrix.

Following a similar framework, Xuan et al. [2017] proposed a multi-label learning model which applies the summation model similar to the one introduce in Eq. (3.73) to incorporate binary features of samples.

Extending the model of Bi and Kwok [2014], Jain et al. [2017] developed a model with the introduction of an exposure variable for each label and sample, detailed as follows:

$$\begin{aligned}
\phi_l &\sim \mathcal{N}(-, -), && (3.83)\\
\boldsymbol{\theta}_i &\sim \mathcal{N}\left(\mathbf{W}^\top \boldsymbol{x}_i, \boldsymbol{\Omega}\right), && (3.84)\\
\epsilon_{ln} &\sim \text{Bern}(-), && (3.85)\\
y_{ln} &\sim \begin{cases} \text{Bern}(\sigma(\phi_l^\top \boldsymbol{\theta}_i)), & \text{if } \epsilon_{ln} = 1 \\ \delta_0, & \text{otherwise} \end{cases} && (3.86)
\end{aligned}$$

where ϵ_{ln} indicates whether label l is exposed or missing in sample i.

Using a similar framework, Gaure et al. further studied the way of incorporating label-label co-occurrence matrix for the case of zero-shot learning, the general idea of which has been studied in Mensink et al. [2014].

In this thesis research, I have developed an efficient model for samples with binary features in multi-label learning, which can be viewed as a variation to the one in shown in Eq. (3.78). More details of this model will be presented in Chapter 7.

3.4 Summary

This chapter has presented the fundamentals and recent research works in the areas of text analysis, graph analysis, and multi-label learning, with a focus on Bayesian latent factor models. Note that each individual area has attracted extensive research interests. For comprehensive reviews of the methods in these areas other than BLFMs, the readers are suggested to refer to the review articles in each individual area. From Chapter 4 to 7, I will present my thesis research in the above areas.

Chapter 4

Bayesian Latent Factor Models for Relational Graph Analysis

Bayesian random graph models [Miller et al., 2009, Zhou, 2015, Caron and Fox, 2017] have been successfully used in relational graph analysis on the tasks of community detection and link prediction. In Bayesian random graph models, a graph is presented as an adjacency matrix, which can be factorised by BLFMs with the latent representations of the nodes. A link between two nodes is generated according to the interactions of their latent representations.

This research is about leveraging node attributes, such as user profiles of a social network and author research interests in a bibliographic graph so as to obtain good community detection and link prediction performance when only a tiny proportion of the links are observed. This is the case where many existing models cannot perform well. In the paper of Zhao et al. [2017a], an effective Bayesian random graph model is proposed, which regresses the latent representations of a node on its attributes, capturing the effect that nodes with similar attributes are likely to be assigned to same communities. The elaborated model structure also facilitates an efficient learning algorithm that utilises the sparsity of both graphs and node attributes. The proposed model achieves the state-of-the-art link prediction results, especially with highly incomplete relational graphs.

The framework of the above model is shown in Figure 4.1, which can be viewed as the extension of the basic framework of BLFM shown in Figure 2.1 in Section 2.2.6 of Chapter 2. Specifically, the proposed model factorises the adjacency matrix of a graph into two latent matrices: the factor loading matrix (the embeddings of the nodes) and the factor correlation matrix, where the factor loading matrix is informed by the node attributes.

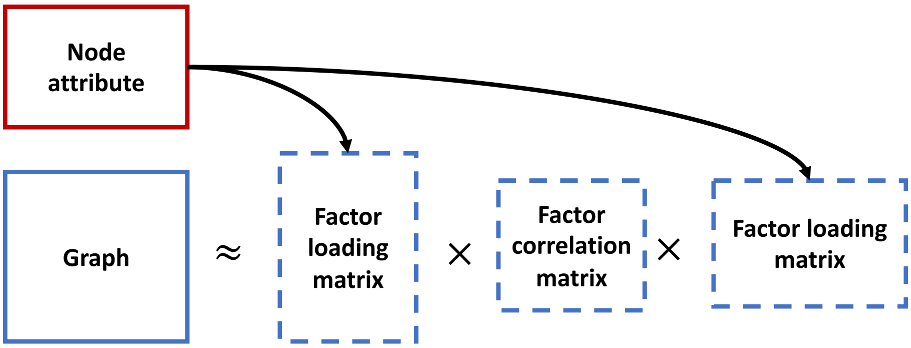


Figure 4.1: Model framework of Zhao et al. [2017a]. The blue rectangles with solid lines and dash lines are the data matrix (the adjacency matrix of a graph) and the latent matrices, respectively. The red rectangle is the matrix of meta-data (the node attributes).

The proposed model works with attributes that are formulated in binary format. However, the binarisation process of non-binary attributes will necessarily cause information loss. Therefore, a possible direction of future research is how to incorporate arbitrary node attributes yet keep the efficiency of the approach. In addition, how the proposed incorporation of node attributes improves the efficiency of MCMC sampling is studied, but whether/how it facilities other inference algorithms such as variational inference is still unknown, which can be a potential future direction as well.

The major content of this chapter is in the following attached paper:

- **H. Zhao**, L. Du, W. Buntine, "Leveraging Node Attributes for Incomplete Relational Data", *International Conference on Machine Learning* (**ICML**) 2017.

The code of this research is released at `https://github.com/ethanhezhao/NARM`.

Leveraging Node Attributes for Incomplete Relational Data

He Zhao[1] Lan Du[1] Wray Buntine[1]

Abstract

Relational data are usually highly incomplete in practice, which inspires us to leverage side information to improve the performance of community detection and link prediction. This paper presents a Bayesian probabilistic approach that incorporates various kinds of node attributes encoded in binary form in relational models with Poisson likelihood. Our method works flexibly with both directed and undirected relational networks. The inference can be done by efficient Gibbs sampling which leverages sparsity of both networks and node attributes. Extensive experiments show that our models achieve the state-of-the-art link prediction results, especially with highly incomplete relational data.

1. Introduction

Relational learning from network data, particularly with probabilistic methods, has gained a wide range of applications such as social network analysis (Xiang et al., 2010), recommender systems (Gopalan et al., 2014b), knowledge graph completion (Hu et al., 2016b), and bioinformatics (Huopaniemi et al., 2010). Generally speaking, the goal of relational learning is to discover and analyse latent clusters of entities (i.e., community detection), and predict missing links (i.e., link prediction).

The standard approach for modelling relational data is latent factor analysis via matrix factorisation and its variations. Among the existing approaches, Non-negative Matrix Factorisation (NMF) and the Stochastic Block Model (SBM) are prominent foundational methods. NMF is usually used to model relationships between two sets of entities such as users and movies in collaborative filtering (Mnih & Salakhutdinov, 2008). While developed independently, SBM (Wang & Wong, 1987; Nowicki & Snijders, 2001) can be viewed as an extension of NMF that introduces a block matrix to capture the interactions between latent factors. There have been many Bayesian extensions of these two methods, relaxing the assumptions and/or introducing extra components, such as the Infinite Relational Model (IRM) (Kemp et al., 2006), the mixture membership stochastic block model (MMSB) (Airoldi et al., 2008), and the non-parametric latent feature models (NLFM) (Miller et al., 2009). Poisson Factorisation (PF) (Dunson & Herring, 2005; Zhou et al., 2012), is a popular version of NMF which models count data with convenient statistical properties (Gopalan et al., 2014b; 2015). Combining the ideas of PF and SBM, the infinite Edge Partition Model (EPM) (Zhou, 2015) and its extensions (Hu et al., 2016b) have proven successful for relational networks.

When a network has less data, relational learning becomes more difficult. One extreme case is the *cold-start* problem (Lin et al., 2013; Sedhain et al., 2014; Zhang & Wang, 2015), where a node has no observed links, making suggestion of links for that node even more challenging. In such cases, it is natural to appeal to side information such as node attributes or features. For instance, papers in citation networks are often associated with categories and authors, and users in Facebook or Twitter are often asked to provide information such as age, gender and interests. It is reasonable to assume that nodes having similar attributes are more likely to relate to each other (i.e., homophily, Nickel et al., 2016). Thus, node attributes serve as important complementary information to relational data.

There are few Bayesian probabilistic relational models that are able to leverage side information. For example, NLFM uses a linear regression model to transform the features of each node into a single number, which contributes to link probabilities. However, side information in NLFM cannot directly influence the latent factors, which gives little support for community detection. As an extension of MMSB, the Non-parametric Meta-data Dependent Relational (NMDR) model (Kim et al., 2012) incorporates attributes into the mixed-membership distribution of each node with the logistic-normal transform, which results in non-conjugacy for inference. Fan et al. (2016) further developed this idea in the Node information Involved Mixture Membership model (niMM), where side information is integrated in a conjugate way. Although these models demonstrate improvement using side information, they

[1] Faculty of Information Technology, Monash University, Australia. Correspondence to: He Zhao <he.zhao@monash.edu>.

Proceedings of the 34

scale quadratically in the number of nodes and the incorporation of side information is often complicated.

Several recent methods (Gopalan et al., 2014a; Acharya et al., 2015; Hu et al., 2016a) extend PF with side information using the additivity of the Poisson and gamma distributions/processes. With improved scalability, the Structural Side Information Poisson Factorisation (SSI-PF) (Hu et al., 2016a) models directed unweighted networks with node labels, such as citation networks with papers labelled with one of several categories. However, its performance remains untested when a node has multiple attributes. Moreover, undirected networks are not handled by SSI-PF.

In this paper we present the Node Attribute Relational Model (NARM)[1], a fully Bayesian approach that models large, sparse, and unweighted relational networks with arbitrary node attributes encoded in binary form. It works with Poisson gamma relational models to incorporate side information. Specifically, we propose the Symmetric NARM (Sym-NARM) for undirected networks, an extension of EPM (Zhou, 2015) and the Asymmetric NARM (Asym-NARM) for directed networks, an extension of PF (Zhou et al., 2012). The proposed models have several key properties: **(1) Effectively modelling node attributes:** the proposed models are able to achieve improved link prediction performance, especially where training data are limited. **(2) Fully Bayesian and conjugate:** the inference is done by efficient, closed-form Gibbs sampling which scales linearly in the number of observed links and takes advantage of the sparsity of node attributes. It makes our models scalable for large but sparse relational networks with large sets of node attributes. **(3) Flexibility**: the proposed models work on directed and undirected relational networks with flat and hierarchical node attributes.

2. The Node Attribute Relational Model

Here we focus on modelling unweighted networks that can be either directed (i.e., the relationship is asymmetric) or undirected. Assume a relational network with N nodes is stored in a binary adjacency matrix $\mathbf{Y} \in \{0,1\}^{N\times N}$ where $y_{i,j} = 1$ indicates the presence of a link between nodes i and j. If the relationship described in the network is symmetric, then $y_{i,j} = y_{j,i}$, and if asymmetric, possibly $y_{i,j} \neq y_{j,i}$. Node attributes are encoded in a binary matrix $\mathbf{F} \in \{0,1\}^{N\times L}$, where L is the total number of attributes. Attribute $f_{i,l} = 1$ indicates attribute l is active with node i and vice versa. Although our models incorporate binary attributes, categorical attributes and real-valued attributes can be converted into binary values with proper transformations (Kim et al., 2012; Fan et al., 2016; Hu et al., 2016a).

2.1. The Symmetric Node Attribute Relational Model

Sym-NARM works with undirected networks. Its generative process is shown in Figure 1. Instead of modelling the binary matrix $\mathbf{Y}$ directly, it applies the Bernoulli-Poisson link (BPL) function (Zhou, 2015) using an underlying latent count matrix $\mathbf{X}$. One first draws a latent count $x_{i,j}$ from the Poisson distribution and then thresholds it at 1 to generate a binary value $y_{i,j}$. This is shown in Eqs. (1)-(3). Analysed in (Zhou, 2015; Hu et al., 2016b;a), BPL has the appealing property that if $y_{i,j} = 0$, then $x_{i,j} = 0$ with probability one. Thus, only non-zeros in $\mathbf{Y}$ need to be sampled, giving huge computational savings for large sparse networks, illustrated in Section 3 and Section 5.4.

The latent matrix $\mathbf{X}$ is further factorised into K latent factors with a non-negative bilinear model: $\mathbf{X} \sim \mathrm{Poi}(\mathbf{\Phi}\mathbf{\Lambda}\mathbf{\Phi}^T)$ where $\mathbf{\Phi} \in \mathbb{R}_+^{N\times K}$ and $\mathbf{\Lambda} \in \mathbb{R}_+^{K\times K}$. $\mathbf{\Phi}$ is referred to as the *node factor loading matrix* where $\phi_{i,k}$ models the strength of the connection between node i and latent factor k. As in SBM, the correlations of the latent factors are modelled in a symmetric matrix $\mathbf{\Lambda}$, referred to as the *block matrix*. Following (Zhou, 2015), we draw $\mathbf{\Lambda}$ from a hierarchical relational gamma process (implemented with truncation as a vector of gamma variables) , shown in Eqs. (8) and (9).

One appealing aspect of our model is the incorporation of node attributes on the prior of $\phi_{i,k}$ (i.e., $g_{i,k}$). Shown in Eq. (5), $g_{i,k}$ is constructed with a log linear combination of $f_{i,l}$. $h_{l,k}$ is referred to as the k^{th} *attribute factor loading* of attribute l, which influences $g_{i,k}$ iff attribute l is active with node i (i.e., $f_{i,l} = 1$). b_k acts as an attribute-free bias for each latent factor k. $h_{l,k}$ and b_k are gamma distributed with mean 1, hence if attribute l does not contribute to latent factor k or is less useful, $h_{l,k}$ is expected to be near 1 and to have little influence on $g_{i,k}$. The hyper-parameter μ_0 controls the variation of $h_{l,k}$.

The intuition of our model is: if two nodes have more common attributes, their gamma shape parameters will be more similar, with similar node factor loadings, resulting in a larger probability that they relate to each other. Moreover, instead of incorporating the node attributes directly into the node factor loadings, Sym-NARM uses them as the prior information using Eq. (4), which results in a principled way of balancing the side information and the network data. In addition, different attributes can contribute differently to the latent factors. For example, the gender of an author may be much less important to co-authorship with others than the research fields. This is controlled by the attribute factor loading $h_{l,k}$ in our model.

2.2. The Asymmetric Node Attribute Relational Model

Extending the Beta Gamma Gamma Poisson factorisation (BGGPF) (Zhou et al., 2012), Asym-NARM works on di-

[1] Code available at `https://github.com/ethanhezhao/NARM/`

$$
\begin{aligned}
y_{i,j} &= \mathbf{1}_{(x_{i,j}>0)} && (1)\\
x_{i,j} &= \sum_{k_1,k_2=1}^{K} x_{i,k_1,k_2,j} && (2)\\
x_{i,k_1,k_2,j} &\sim \mathrm{Poi}(\phi_{i,k_1}\lambda_{k_1,k_2}\phi_{j,k_2}) && (3)\\
\phi_{i,k} &\sim \mathrm{Ga}(g_{i,k}, 1/c_i) && (4)\\
g_{i,k} &= b_k \prod_{l=1}^{L} h_{l,k}^{f_{i,l}} && (5)\\
h_{l,k} &\sim \mathrm{Ga}\left(\mu_0, 1/(1/\mu_0)\right) && (6)\\
b_k &\sim \mathrm{Ga}\left(\mu_0, 1/(1/\mu_0)\right) && (7)\\
\lambda_{k_1,k_2} &\sim \begin{cases} \mathrm{Ga}(\epsilon r_k, 1/a_0), & \text{if } k_1 = k_2 = k\\ \mathrm{Ga}(r_{k_1} r_{k_2}, 1/a_0), & \text{otherwise} \end{cases} && (8)\\
r_k &\sim \mathrm{Ga}(\gamma_0/K, 1/c_0) && (9)
\end{aligned}
$$

Figure 1. The generative model of Sym-NARM. $\mathbf{1}_{(\cdot)}$ is the indicator function. $\mathrm{Poi}(\cdot)$ and $\mathrm{Ga}(\cdot,\cdot)$ stand for the Poisson distribution and the gamma distribution respectively. Conjugate gamma priors are imposed on the hyper-parameters: γ_0, ϵ, c_0, c_i, and a_0.

$$
\begin{aligned}
y_{i,j} &= \mathbf{1}_{(x_{i,j}>0)} && (10)\\
x_{i,j} &\sim \sum_{k}^{K} x_{i,j,k} && (11)\\
x_{i,j,k} &\sim \mathrm{Poi}(\phi_{i,k}\theta_{j,k}) && (12)\\
\phi_{i,k} &\sim \mathrm{Ga}\left(g_{i,k}, \frac{q_k}{1-q_k}\right) && (13)\\
q_k &\sim \mathrm{Be}\left(c_0\epsilon, c_0(1-\epsilon)\right) && (14)\\
g_{i,k} &= b_k \prod_{l=1}^{L} h_{l,k}^{f_{i,l}} && (15)\\
h_{l,k} &\sim \mathrm{Ga}\left(\mu_0, 1/(1/\mu_0)\right) && (16)\\
b_k &\sim \mathrm{Ga}\left(\mu_0, 1/(1/\mu_0)\right) && (17)\\
\theta_{:,k} &\sim \mathrm{Dir}_N(a_0\vec{1}) && (18)
\end{aligned}
$$

Figure 2. The generative model of Asym-NARM. $\mathrm{Dir}_N(\cdot)$ and $\mathrm{Be}(\cdot,\cdot)$ stand for the N dimensional Dirichlet distribution and the beta distribution respectively. $\mu_0, \nu_0, a_0, e_0, f_0, c_0, \epsilon$ are the hyper-parameters.

rected relational networks with node attributes incorporated in a similar way to Sym-NARM. Figure 2 shows its generative process. Here the latent count matrix $\mathbf{X}$ is factorised as $\mathbf{X} \sim \mathrm{Poi}(\mathbf{\Phi\Theta})$, where $\mathbf{\Phi} \in \mathbb{R}_+^{N\times K}$ and $\mathbf{\Theta} \in \mathbb{R}_+^{K\times N}$ are referred to as the *factor loading matrix* and the *factor score matrix* respectively. Similar to SSI-PF, the node attributes are incorporated on the prior of $\mathbf{\Phi}$.

2.3. Incorporating Hierarchical Node Attributes

Relational networks can be associated with hierarchical side information (Hu et al., 2016a). For example, in a patent citation network, patents can be labelled with the International Patent Classification (IPC) code, which is a hierarchy of patent categories and sub-categories. Suppose the second level attributes are stored in a binary matrix $\mathbf{F}' \in \{0,1\}^{L\times M}$ where M is the number of attributes in the second level. Our models can be used to incorporate hierarchical node attributes via a straightforward extension: replace hyper-parameter μ_0 in Eq. (6) with $\mu_{l,k} = \prod_m^M \delta_{m,k}^{f'_{l,m}}$. This extension mirrors what is done for first level attributes.

3. Inference with Gibbs Sampling

Both Sym-NARM and Asym-NARM enjoy local conjugacy so the inference of all latent variables can be done by closed-form Gibbs sampling. Moreover, the inference only needs to be conducted on the non-zero entries in $\mathbf{Y}$ and $\mathbf{F}$. This section focuses on the sampling of $h_{l,k}$ (b_k), the key variable in the proposed incorporation of node attributes. The sampling of the other latent variables is similar to those in EPM and BGGPF, detailed in (Zhou, 2015; Zhou et al., 2012). As the sampling for $h_{l,k}$ is analogous in Sym-NARM and Asym-NARM, our introduction will be based on Asym-NARM alone.

With the Poisson gamma conjugacy, the likelihood for $g_{i,k}$ with $\phi_{i,k}$ marginalised out is:

$$\mathrm{p}(g_{i,k} \mid x_{i,\cdot,k}) \propto (1-q_k)^{g_{i,k}} \frac{\Gamma(g_{i,k}+x_{i,\cdot,k})}{\Gamma(g_{i,k})} \qquad (19)$$

where $x_{i,\cdot,k} = \sum_j x_{i,j,k}$ and $x_{i,j,k}$ is the latent count. The gamma ratio in Eq. (19), i.e., the Pochhammer symbol for a rising factorial, can be augmented with an auxiliary variable $t_{i,k}$: $\frac{\Gamma(g_{i,k}+x_{i,\cdot,k})}{\Gamma(g_{i,k})} = \sum_{t_{i,k}=0}^{x_{i,\cdot,k}} S_{t_{i,k}}^{x_{i,\cdot,k}} g_{i,k}^{t_{i,k}}$ where S_t^x indicates an unsigned Stirling number of the first kind (Chen et al., 2011; Teh et al., 2012; Zhou & Carin, 2015).

Taking $\mathcal{O}(x_{i,\cdot,k})$, $t_{i,k}$ can be directly sampled by a Chinese Restaurant Process with $g_{i,k}$ as the concentration and $x_{i,\cdot,k}$ as the number of customers:

$$t_{i,k} \leftarrow t_{i,k} + \mathrm{Bern}\left(\frac{g_{i,k}}{g_{i,k}+i'}\right) \text{ for } i' = 1 : x_{i,\cdot,k} \qquad (20)$$

where $\mathrm{Bern}(\cdot)$ is the Bernoulli distribution. Alternatively, for large $x_{i,\cdot,k}$, because the standard deviation of $t_{i,k}$ is $\mathcal{O}(\sqrt{\log x_{i,\cdot,k}})$ (Buntine & Hutter, 2012), one can sample $t_{i,k}$ in a small window around the current value (Du et al., 2010).

With the above augmentation and Eq. (15), we get:

$$\mathrm{p}(\mathbf{G},\mathbf{H} \mid x_{:,\cdot,:},\mathbf{T},\mathbf{F}) \propto \qquad (21)$$
$$\prod_{i=1}^{N}\prod_{k=1}^{K} S_{t_{i,k}}^{x_{i,\cdot,k}} e^{-\log\left(\frac{1}{1-q_k}\right) g_{i,k}} \cdot \prod_{l=1}^{L}\prod_{k=1}^{K} h_{l,k}^{\sum_{i=1}^{N} f_{i,l} t_{i,k}}$$

Recall that all the attributes are binary and $h_{l,k}$ influences $g_{i,k}$ only when $f_{i,l} = 1$. Extracting all the terms related to

$h_{l,k}$ in Eq. (21), we get the likelihood of $h_{l,k}$:

$$\mathrm{p}\left(h_{l,k} \,\middle|\, \frac{g_{i,k}}{h_{l,k}}, t_{:,k}, f_{:,l}\right) \propto \tag{22}$$
$$e^{-h_{l,k} \log\left(\frac{1}{1-q_k}\right) \sum_{i=1:f_{i,l}=1}^{N} \frac{g_{i,k}}{h_{l,k}}} h_{l,k}^{\sum_{i=1}^{N} f_{i,l} t_{i,k}}$$

where $\frac{g_{i,k}}{h_{l,k}}$ is the value of $g_{i,k}$ with $h_{l,k}$ removed when $f_{i,l} = 1$. The likelihood function above is in a form that is conjugate to the gamma prior. Therefore, it is straightforward to yield the following sampling strategy for $h_{l,k}$:

$$h_{l,k} \sim \mathrm{Ga}(\mu', 1/\nu') \tag{23}$$

$$\mu' = \mu_0 + \sum_{i=1:f_{i,l}=1}^{N} t_{i,k} \tag{24}$$

$$\nu' = 1/\mu_0 - \log(1 - q_k) \sum_{i=1:f_{i,l}=1}^{N} \frac{g_{i,k}}{h_{l,k}} \tag{25}$$

Precomputed with Eq. (15), $g_{i,k}$ can be updated with Eq. (26), after $h_{l,k}$ is sampled.

$$g_{i,k} \leftarrow \frac{g_{i,k} h'_{l,k}}{h_{l,k}} \text{ for } i = 1 : N \text{ and } f_{i,l} = 1 \tag{26}$$

where $h'_{i,k}$ is the newly sampled value of $h_{i,k}$.

To compute Eqs. (24)-(26), we only need to iterate over the nodes that attribute l is active with (i.e., $f_{i,l} = 1$). Thus, the sampling for $\mathbf{H}$ takes $\mathcal{O}(D'KL)$ where D' is the average number of nodes that an attribute is active with. This demonstrates how the sparsity of node attributes is leveraged. As the mean of $x_{i,\cdot,k}$ is D/K, sampling the tables $\mathbf{T} \in \mathbb{N}^{N \times K}$ takes $\mathcal{O}(ND)$ which can be accelerated with the window sampling technique explained above.

We show the computational complexity of our and related models in Table 1. The empirical comparison of running speed is in Section 5.4. By taking advantage of both network sparsity and node attribute sparsity, our models are more efficient than the competitors, especially on large sparse networks with large sets of attributes.

4. Related work

Compared with the node-attribute models such as NMDR and niMM whose methods result in complicated inference, our Sym-NARM is much more efficient on large sparse networks, illustrated in Table 1.

The most closely related model to our Asym-NARM, also extending the BGGPF algorithm, is SSI-PF. But it uses the gamma additivity to construct the prior of node factor loadings with the sum of attribute factor loadings. Our model has several advantages over SSI-PF: (1) The derivation of Gibbs sampling of SSI-PF requires that each column of $\mathbf{\Theta}$ is normalised (Eq. (18)). This limits the application of SSI-PF to other models such as EPM which is an unnormalised model. (2) Shown in Table 1, Asym-NARM enjoys more efficient computational complexity. (3) Shown in Section 5, our model is more effective especially when a node has multiple attributes.

Table 1. The computational complexity for the compared models. N: number of nodes. K: number of latent factors. L: number of node attributes. D: the average degree (number of edges) per node ($D \ll N$ in sparse networks). D': the average number of nodes that an attribute is active with (usually, $D' < N$). For the models that incorporate node attributes (marked with a *), the complexity with one level attributes is shown.

Model	Complexity
Models with the block matrix	
*NMDR (Kim et al., 2012)	$\mathcal{O}(N^2K + NKL)$
*niMM (Fan et al., 2016)	$\mathcal{O}(N^2K^2 + NKL)$
EPM (Zhou, 2015)	$\mathcal{O}(NK^2D)$
***Sym-NARM**	$\mathcal{O}(NK^2D + D'KL)$
Models without the block matrix	
BGGPF (Zhou et al., 2012)	$\mathcal{O}(NKD)$
*SSI-PF (Hu et al., 2016a)	$\mathcal{O}(NKDL)$
***Asym-NARM**	$\mathcal{O}(NKD + D'KL)$

There are also models that extend PF and collective matrix factorisation (Singh & Gordon, 2008) to jointly factorise relational networks and document-word matrices such as (Gopalan et al., 2014a; Zhang & Wang, 2015; Acharya et al., 2015). Our NARM models incorporate general node attributes (not only texts) as the priors of the factor loading matrix in a supervised manner, rather than jointly modelling the side information in an unsupervised manner.

Another related area is supervised topic models such as (Mcauliffe & Blei, 2008; Ramage et al., 2009; Lim & Buntine, 2016). The Dirichlet Multinomial Regression (DMR) model (Mimno & McCallum, 2012) is the most related one to ours. It models document attributes on the priors of the topic proportions with the logistic-normal transform. For comparison, we propose DMR-MMSB, extending MMSB with the DMR technique to incorporate side information on the mixed-membership distribution of each node.

5. Experiments

In this section we evaluate Sym-NARM and Asym-NARM with a set of the link prediction tasks on 10 real-world relational datasets with different sizes and various kinds of node attributes. We compare our models with the state-of-the-art relational models, demonstrating that our models outperform the competitors on those datasets in terms of link prediction performance and per-iteration running time. We report the average area under the curve of both the receiver operating characteristic (AUC-ROC) and precision recall (AUC-PR) for quantitatively analysing the models. Moreover, we perform qualitative analysis by comparing the link probabilities estimated by the compared models.

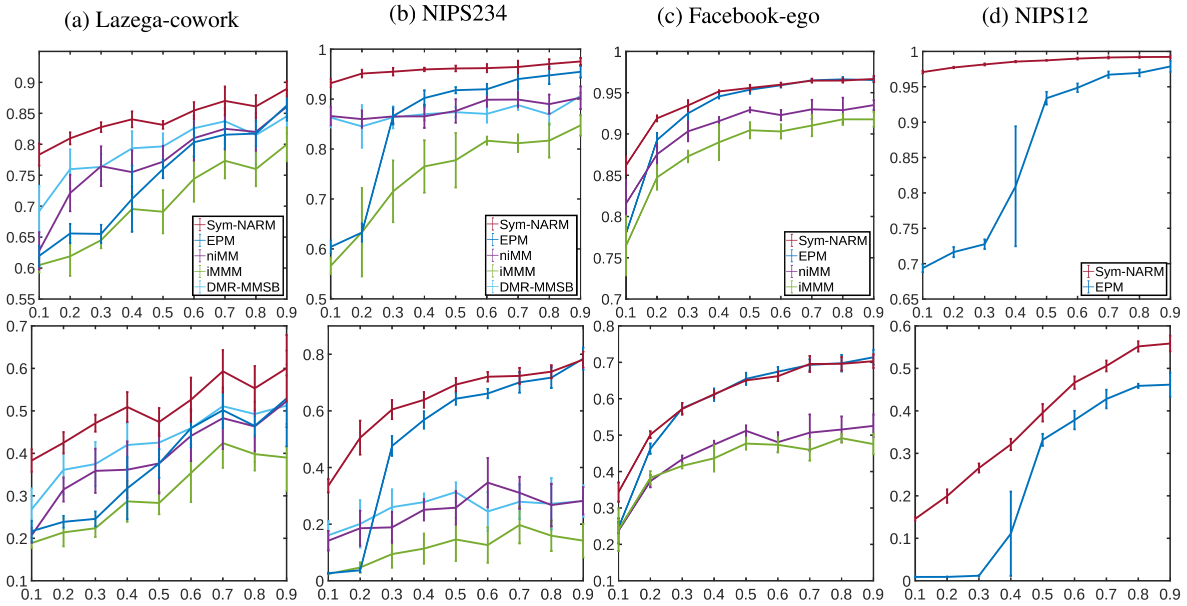


Figure 3. The AUC-ROC (the first row) and AUC-PR (the second row) scores on the undirected networks. The values on the horizontal axis are the proportions of the training data and each of the error bars is the standard deviation over the five random splits for one proportion. DMR-MMSB achieves its best performance at $K = 5$ and 10 on Lazega-cowork and NIPS234 respectively.

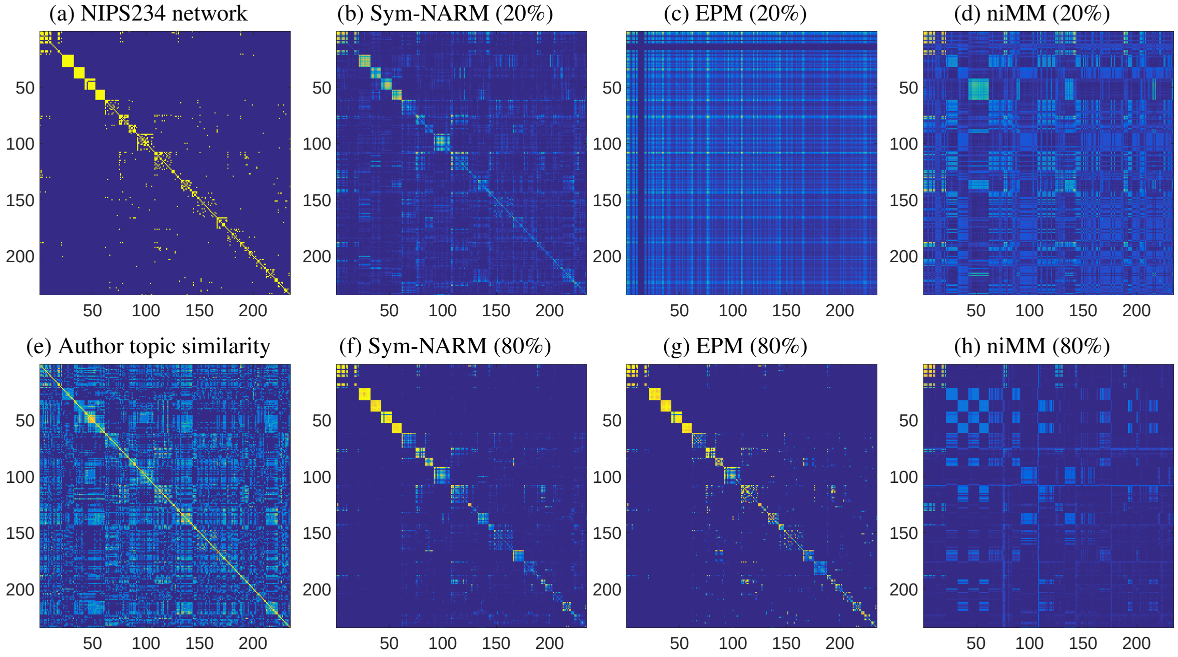


Figure 4. The link probability estimations in NIPS234. Similar to (Zhou, 2015), the nodes are reordered to make a node with a larger index belong to the same or a smaller-size community, where the disjoint community assignments are obtained by analysing the results of Sym-NARM. (a) The original NIPS234 network. (e) The topic similarity of the authors, obtained by the pairwise cosine distances of the topic proportions, with a brighter colour representing a closer distance. (b)-(d) and (f)-(h) Estimated link probabilities with 20% and 80% training data respectively for each compared model.

5.1. Link Prediction on Undirected Networks

For the link prediction task on undirected network data, we compared our **Sym-NARM** with two models that do not consider node attributes, **EPM** (Zhou, 2015), a state-of-the-art relational model, and **iMMM** (Koutsourelakis & Eliassi-Rad, 2008), a non-parametric version of MMSB, and two node attribute models, **niMM** (Fan et al., 2016), a non-parametric relational model which has been demonstrated to outperform NMDR (Kim et al., 2012), and **DMR-MMSB**, our extension to MMSB using the Dirichlet Multinomial Regression (Mimno & McCallum, 2012). Sym-NAMR was implemented in MATLAB on top of the EPM code and we used the code released by the original authors for EPM and niMM. iMMM was implemented by Fan et al. (2016) as a variant of niMM.

The description of the four datasets used is given below:

- **Lazega-cowork:** This dataset (Lazega, 2001) contains 378 links of the co-work relationship among 71 attorneys. Each attorney is associated with attributes such as gender, office location, and age. After discretisation and binarisation, we derived a 71 × 18 binary node attribute matrix with 497 non-zero entries.
- **NIPS234:** This is a co-author network of the 234 authors with 598 links extracted from NIPS 1-17 conferences (Zhou, 2015). We merged all the papers written by the same author as a document, and then trained a LDA model with 100 topics. The 5 most frequent topics were used as the attributes, which gives us a 234 × 100 attribute matrix with 1170 non-zero entries.
- **Facebook-ego:** The original dataset (McAuley & Leskovec, 2012) was collected from survey participants of Facebook users. Out of the 10 circles (i.e., friend lists), we used the first circle that contains 347 users with 2519 links. Each user is associated with 227 binary attributes, encoding side information such as age, gender, and education. We got a 347 × 227 binary node attribute matrix with 3318 non-zero entries.
- **NIPS12:** NIPS12 was collected from NIPS papers in vols 0-12. It is a median-size co-author network with 2037 authors and 3134 links. Similar to NIPS234, we used the 5 most frequent topics as the attributes for each author. We got a 2037 × 100 binary node attribute matrix with 10185 non-zero entries.

5.1.1. Experimental Settings

For each dataset, we varied the training data from 10% to 90% and used the remaining in testing. For each proportion, to generate five random splits, we used the code in the EPM package (Zhou, 2015) which splits a network in terms of its nodes. The reported AUC-ROC/PR scores were averaged over the five splits. We used the default hyper-parameter settings enclosed in the released code for EPM, niMM and iMMM. For our Sym-NARM, we set $\mu_0 = 1$ and all the other hyper-parameters the same as those in EPM. Note that the models in comparison except DMR-MMSB are non-parametric models. For Sym-NARM and EPM, we set the truncation level large enough for each dataset: $K_{max} = 50, 100, 256$ for Lazega-cowork, Facebook-ego and NIPS234, NIPS12 respectively. For DMR-MMSB, we varied K in $\{5, 10, 25, 50\}$ and reported the best one. Following (Zhou, 2015), we used 3000 MCMC iterations and computed AUC-ROC/PR with the average probability over the last 1500. The performance of iMMM and niMM on NIPS12 and DMR-MMSB on Facebook-ego and NIPS12 are not reported as the datasets are too large for them given our computational resources.

5.1.2. Results

The AUC-ROC/PR scores are reported in Figure 3. Overall, our Sym-NARM model performs significantly better than niMM, iMMM, and DMR-MMSB on all the datasets, and EPM on 3 datasets (except Facebook-ego with large training proportions). It is interesting that the performance of EPM on Facebook-ego gradually approaches ours when more than 30% training data were used. Note that Facebook-ego is much denser than the others, which means the network information itself could be rich enough for EPM to reconstruct the network and the node attributes contribute less. However in general, when relational data are highly incomplete (with less training data), our model is able to achieve improved link prediction performance.

To illustrate how side information helps, we qualitatively compared our model with EPM and niMM by estimating the link probabilities on NIPS234, shown in Figure 4. With 20% training data, EPM does not give a meaningful reconstruction of the original network, but it starts to with more data presented. The similarity of the authors' topics in Figure 4e matches the original network, demonstrating the usefulness of the topics, but with some error. Using the topics as the authors' attributes, our Sym-NARM achieves reasonably good reconstruction of the network with only 20% training data, further improving with 80% training data. Although niMM uses the same node attributes, its performance is not as good and is even outperformed by EPM with 80% training data.

5.2. Link Prediction on Directed Networks

Here we compared our **Asym-NARM** (implemented in MATLAB on top of the BGGPF code) with two models that do not consider node attributes, **BGGPF** (Zhou et al., 2012) and **iMMM**, and three node-attribute models, **niMM**, **SSI-PF** (Hu et al., 2016a) and **DMR-MMSB**. We used the following four datasets:

- **Lazega-advice:** This dataset is a directed network with 892 links of the advice relation among the attorneys. The node attributes are the same as in Lazega-cowork.
- **Citeseer:** This dataset[2] contains a citation network with

[2] http://linqs.umiacs.umd.edu/projects/ /projects/lbc/index.html

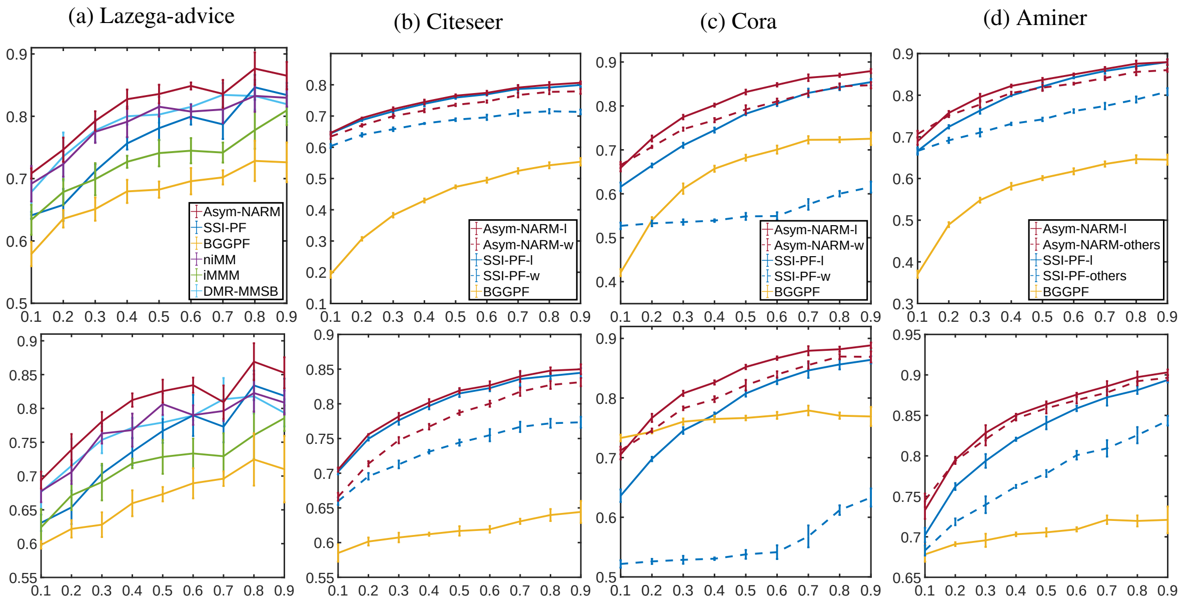


Figure 5. The AUC-ROC (the first row) and AUC-PR (the second row) scores on the directed networks. The models with "-l" and "-w" use the labels and the words as attributes respectively. The models with "-others" in Aminer use the extra attributes. DMR-MMSB achieves its best performance at $K = 10$ on Lazega-advice.

4591 links of 3312 papers, labelled with one of 6 categories. For each paper, we used both the category label and the presence/absence of 500 most frequent words as two separate attribute sets. We got a 3312×500 word attribute matrix with 65674 non-zero entries.

- **Cora:** This dataset[2] contains a citation network with 5429 links of 2708 papers in machine learning, labelled with one of 7 categories. Similar to Citeseer, we used both the category label and the 500 most frequent words as two separate attribute sets. We got a 2708×500 word attribute matrix with 39268 non-zero entries.
- **Aminer:** The Aminer dataset (Tang et al., 2009) contains a citation network with 2555 papers labelled with 10 categories and 5967 links. We further collected information of each paper via the Aminer's API, including the authors' names (2597 unique authors), abstract, venue, year, and number of citations. For the abstract, we extract the 5 most frequent topics for each paper in a similar way to NIPS234. In total, we prepared two sets of attributes: the labels and the others formed with the combination of all collected information.

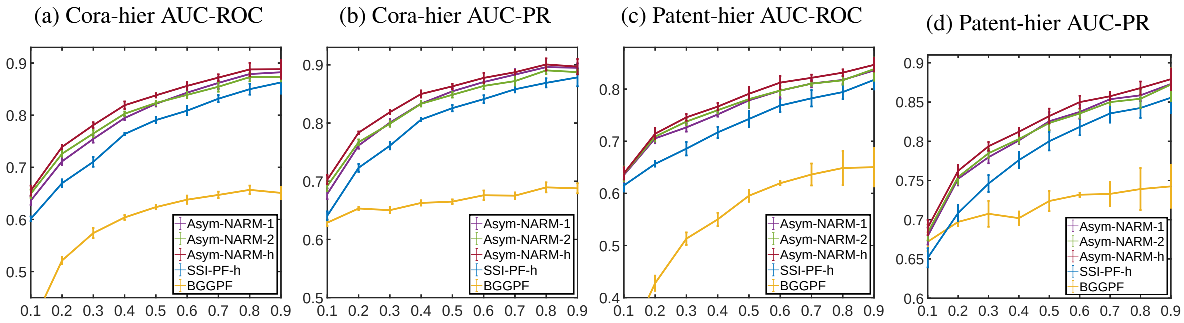
5.2.1. Experimental Settings

For fair comparison, we generated training/testing data with the code in the SSI-PF package, which splits a network in terms of its links. We used the default hyper-parameter settings of BGGPF, SSI-PF, and niMM, provided by the original authors. K_{max} was set to 50 on Lazega-advice and 200 (same as (Hu et al., 2016a)) on all the other three datasets. For our Asym-NARM, we set $\mu_0 = 1$ and the other hyper-parameters the same as those used in (Zhou et al., 2012; Hu et al., 2016a). Following the suggestion of Hu et al. (2016a), we used 1500 MCMC iterations in total and the last 500 samples to compute the AUC-ROC/PR scores. Since Citeseer, Cora, and Aminer are already too large for niMM, iMMM, and DMR-MMSB to produce results in reasonable time given our computational resources, we reported their performance only on Lazega-advice.

5.2.2. Results

Shown in Figure 5a, Asym-NARM gains better results in terms of AUC-ROC/PR on Lazega-advice in most of the training proportions. Overall, the node-attribute models perform better than the models that do not consider node attributes, showing the usefulness of node attributes. On the other three datasets, we used different sets of attributes to study how different attributes influence the performance of Asym-NARM and SSI-PF.

In general, Asym-NARM performs better than SSI-PF regardless of which set of attributes is used. The performance of SSI-PF approaches ours in Citeseer with the labels as attributes (indicated by "-l"). But the gap between SSI-PF and our model becomes larger when the words are used as attributes (indicated by "-w"). In Cora, SSI-PF with the words does not perform as well as its non-node-attribute counterpart, BGGPF, indicating it may not be as robust as our model with large sets of attributes. To investigate this, we varied the number of the most frequent words from 10 to 500 for Asym-NARM and SSI-PF on Citeseer and Cora. With more words, the AUC-ROC/PR score of SSI-PF de-



(a) Cora-hier AUC-ROC (b) Cora-hier AUC-PR (c) Patent-hier AUC-ROC (d) Patent-hier AUC-PR

Figure 6. The AUC-ROC and AUC-PR scores on the networks with hierarchical attributes. The models with the first level attributes only, the second level attributes only, and the hierarchical attributes are marked with “-1”, “-2”, and “-h” respectively.

grades increasingly. We further checked the prior of the node factor loadings in SSI-PF (the variable that incorporates node attributes and corresponds to $g_{i,k}$ in our model) and found that the coefficient of variation of each node’s prior drops dramatically, indicating with more words, SSI-PF is failing to use the supervised information in the words.

5.3. Link Prediction with Hierarchical Node Attributes

Here we used two datasets with hierarchical node attributes: (1) **Cora-hier**: a citation network with 1712 papers and 6308 links extracted from the original Cora dataset[3]. The papers are labelled with one of 63 sub-areas (first level) and each sub-area belongs to one of 10 primary areas (second level), such as “machine learning in artificial intelligence” and “memory management in operating systems”; (2) **Patent-hier**: a citation network with 1461 patents and 2141 links from the National Bureau of Economic Research where the hierarchical International Patent Classification (IPC) code of a patent is used as attributes.

The AUC-ROC/PR scores in Figure 6 show that our Asym-NARM with hierarchical attributes outperforms the others, which demonstrates leveraging hierarchical side information is beneficial to link prediction. Although SSI-PF also models the hierarchical attributes, its performance in these two datasets is not comparable with our model’s.

5.4. Running Time

In this section, we compare the running time of the models for directed networks (all implemented in MATLAB and running on a desktop with 3.40 GHz CPU and 16GB RAM). Using 80% data for training, the running time for Asym-NARM, SSI-PF, and niMM on Aminer with different sets of node attributes is reported in Table 2. Note DMR-MMSB did not complete with “Authors” and “All” due to our computational resources. Asym-NARM is about 10 times faster than SSI-PF with all the attributes and about 2 times faster with the labels. Thus Asym-NARM is more efficient, especially with large sets of attributes, supporting the complexity analysis in Table 1.

Table 2. The running time (seconds per iteration) of the compared models on Aminer. AT: the topics extracted from the abstracts. All: the combination of all the attributes we have.

Attr	Non-zeros & attr size	**Asym-NARM**	SSI-PF	niMM	DMR-MMSB $K = 50$
Label	2660 2555*10	**0.26**	0.48	134.11	89.12
AT	12775 2555*100	**0.29**	0.87	135.22	126.44
Authors	5647 2555*2597	**0.33**	2.99	136.41	-
All	31273 2555*3058	**0.51**	5.21	136.14	-

6. Conclusion

As a summary of the experiments, Asym/Sym-NARM achieved better link prediction performance with faster inference. While EPM, a non-node-attribute model, performed well on nearly complete networks, it degraded with less training data. niMM and DMR-MMSB, extensions to MMSB with the logistic-normal transform, had similar results to Sym-NARM but scaled inefficiently. SSI-PF’s performance and scalability were not as good as Asym-NARM in the presented cases with flat and hierarchical attributes and it was less effective with larger numbers of attributes.

Thus NARM is a comparatively simple yet effective and efficient way of incorporating node attributes, including hierarchical attributes, for relational models with Poisson likelihood. This leads to improved link prediction and matrix completion for less complete relational data of both directed and undirected networks. With the efficient inference, our models can be used to model large sparse relational networks with node attributes.

NARM can easily be extended to multi-relational networks such as (Hu et al., 2016b) and topic models with document and word attributes, which is left for our future work.

[3] `https://people.cs.umass.edu/~mccallum/data.html`

References

Acharya, A., Teffer, D., Henderson, J., Tyler, M., Zhou, M., and Ghosh, J. Gamma process Poisson factorization for joint modeling of network and documents. In *Machine Learning and Knowledge Discovery in Databases, European Conference, Part I*, pp. 283–299. Springer, 2015.

Airoldi, E.M., Blei, D.M., Fienberg, S.E., and Xing, E.P. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9(Sep):1981–2014, 2008.

Buntine, W. and Hutter, M. A Bayesian view of the Poisson-Dirichlet process. *arXiv preprint arXiv:1007.0296v2 [math.ST]*, 2012.

Chen, C., Du, L., and Buntine, W. Sampling table configurations for the hierarchical Poisson-Dirichlet process. In *Machine Learning and Knowledge Discovery in Databases. European Conference, Part I*, pp. 296–311. Springer, 2011.

Du, L., Buntine, W., and Jin, H. A segmented topic model based on the two-parameter Poisson-Dirichlet process. *Machine Learning*, 81:5–19, 2010.

Dunson, D.B. and Herring, A.H. Bayesian latent variable models for mixed discrete outcomes. *Biostatistics*, 6(1): 11–25, 2005.

Fan, X., Xu, D., Yi, R., Cao, L., and Song, Y. Learning nonparametric relational models by conjugately incorporating node information in a network. *IEEE transactions on cybernetics*, 2016.

Gopalan, P., Charlin, L., and Blei, D. Content-based recommendations with Poisson factorization. In *Advances in Neural Information Processing Systems*, pp. 3176–3184, 2014a.

Gopalan, P., Ruiz, F.J., Ranganath, R., and Blei, D.M. Bayesian nonparametric Poisson factorization for recommendation systems. In *17th International Conference on Artificial Intelligence and Statistics*, pp. 275–283, 2014b.

Gopalan, P., Hofman, J.M., and Blei, D.M. Scalable recommendation with hierarchical Poisson factorization. In *31st Conference on Uncertainty in Artificial Intelligence*, pp. 326–335, 2015.

Hu, C., Rai, P., and Carin, L. Non-negative matrix factorization for discrete data with hierarchical side-information. In *19th International Conference on Artificial Intelligence and Statistics*, pp. 1124–1132, 2016a.

Hu, C., Rai, P., and Carin, L. Topic-based embeddings for learning from large knowledge graphs. In *19th International Conference on Artificial Intelligence and Statistics*, pp. 1133–1141, 2016b.

Huopaniemi, I., Suvitaival, T., Nikkilä, J., Orešič, M., and Kaski, S. Multivariate multi-way analysis of multi-source data. *Bioinformatics*, 26(12):i391–i398, 2010.

Kemp, C., Tenenbaum, J.B., Griffiths, T.L., Yamada, T., and Ueda, N. Learning systems of concepts with an infinite relational model. In *21st National Conference on Artificial Intelligence*, pp. 381–388. AAAI, 2006.

Kim, D.I., Hughes, M., and Sudderth, E. The nonparametric metadata dependent relational model. In *29th International Conference on Machine Learning*, pp. 1559–1566, 2012.

Koutsourelakis, P.-S. and Eliassi-Rad, T. Finding mixed-memberships in social networks. In *AAAI Spring Symposium: Social Information Processing*, pp. 48–53, 2008.

Lazega, E. *The collegial phenomenon: The social mechanisms of cooperation among peers in a corporate law partnership*. Oxford University Press on Demand, 2001.

Lim, K. and Buntine, W. Bibliographic analysis on research publications using authors, categorical labels and the citation network. *Machine Learning*, 103(2):185–213, 2016.

Lin, J., Sugiyama, K., Kan, M.-Y., and Chua, T.-S. Addressing cold-start in app recommendation: Latent user models constructed from Twitter followers. In *36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 283–292, 2013.

McAuley, J.J. and Leskovec, J. Learning to discover social circles in ego networks. In *Advances in Neural Information Processing Systems*, pp. 548–56, 2012.

Mcauliffe, J.D. and Blei, D.M. Supervised topic models. In *Advances in Neural Information Processing Systems*, pp. 121–128, 2008.

Miller, K., Jordan, M.I., and Griffiths, T.L. Nonparametric latent feature models for link prediction. In *Advances in Neural Information Processing Systems*, pp. 1276–1284, 2009.

Mimno, D. and McCallum, A. Topic models conditioned on arbitrary features with Dirichlet-multinomial regression. In *24th Conference on Uncertainty in Artificial Intelligence*, pp. 411–418, 2012.

Mnih, A. and Salakhutdinov, R. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*, pp. 1257–1264, 2008.

Nickel, M., Murphy, K., Tresp, V., and Gabrilovich, E. A Review of Relational Machine Learning for Knowledge Graphs. *Proceedings of the IEEE*, 104(1):11–33, 2016.

Nowicki, K. and Snijders, T.A.B. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001.

Ramage, D., Hall, D., Nallapati, R., and Manning, C.D. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *2009 Conference on Empirical Methods in Natural Language Processing: Volume 1*, pp. 248–256. ACL, 2009.

Sedhain, S., Sanner, S., Braziunas, D., Xie, L., and Christensen, J. Social collaborative filtering for cold-start recommendations. In *8th ACM Conference on Recommender Systems*, pp. 345–348, 2014.

Singh, A.P. and Gordon, G.J. Relational learning via collective matrix factorization. In *14th ACM SIGKDD International Conference on Knowledge discovery and data mining*, pp. 650–658. ACM, 2008.

Tang, J., Sun, J., Wang, C., and Yang, Z. Social influence analysis in large-scale networks. In *15th ACM SIGKDD International Conference on Knowledge discovery and data mining*, pp. 807–816. ACM, 2009.

Teh, Y.W., Jordan, M.I., Beal, M.J., and Blei, D.M. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2012.

Wang, Y.J. and Wong, G.Y. Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82(397):8–19, 1987.

Xiang, R., Neville, J., and Rogati, M. Modeling relationship strength in online social networks. In *19th International Conference on World Wide Web*, pp. 981–990. ACM, 2010.

Zhang, W. and Wang, J. A collective Bayesian Poisson factorization model for cold-start local event recommendation. In *21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1455–1464, 2015.

Zhou, M. Infinite edge partition models for overlapping community detection and link prediction. In *18th International Conference on Artificial Intelligence and Statistics*, pp. 1135–1143, 2015.

Zhou, M. and Carin, L. Negative binomial process count and mixture modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):307–320, 2015.

Zhou, M., Hannah, L., Dunson, D.B., and Carin, L. Beta-negative binomial process and Poisson factor analysis. In *15th International Conference on Artificial Intelligence and Statistics*, pp. 1462–1471, 2012.

Chapter 5

Meta-data Enhanced Topic Models for Text Analysis

BLFMs have been an important series of tools for text analysis by discovering interpretable latent topics. Conventional latent factor models for texts (i.e., topic models) such as Latent Dirichlet Allocation (LDA) [Blei et al., 2003] learn topics purely from the content of a text corpus, ignoring the associated meta-data, such as document-level meta-data like labels, authors, timestamps, and word-level meta-data like word embeddings. This chapter introduces three proposed models on enhancing topic modelling with document and word meta-data so as to get better modelling performance and interpretability.

In the paper of Zhao et al. [2017c], a general topic modelling framework is proposed, which efficiently incorporates both document-level and word-level meta-data in binary form. The intuition of this work is that documents with similar meta-data are likely to discuss similar topics and words having similar meanings (encoded in word embeddings) but different morphological forms, like "dog" and "puppy", are likely to be in the same topic, even if they barely co-occur in the corpus. The proposed model achieves significantly better modelling results and interpretability, especially on short texts such as tweets and news headlines, where meta-data play a more significant role. This framework is well-engineered on MALLET[1], which is able to run efficiently with multiple threads in multi-core machines on large-scale datasets. The details of this research are in the following attached paper:

- **H. Zhao**, L. Du, W. Buntine, G. Liu, "MetaLDA: A Topic Model that Efficiently Incorporates Meta information", long paper in *International Conference on Data Mining* (**ICDM**) 2017.

The paper of Zhao et al. [2018a] is a journal extension to Zhao et al. [2017c], which gives comprehensive studies, comparisons, and discussions on how different parts of the model affect its performance, detailed derivations of the inference algorithm, how different word embeddings affect the performance, how the multi-thread implementation accelerates the learning speeds with more quantitative and qualitative demonstrations. The details of this research are in the following attached paper:

- **H. Zhao**, L. Du, W. Buntine, G. Liu, "Leveraging External Information In Topic Modelling", in *Knowledge and Information Systems* (**KAIS**) 2018.

In conventional topic models, a topic is assumed to be a distribution spreading over all the words in the vocabulary. However, in practice, a topic can only be interpreted with

[1] `http://mallet.cs.umass.edu`

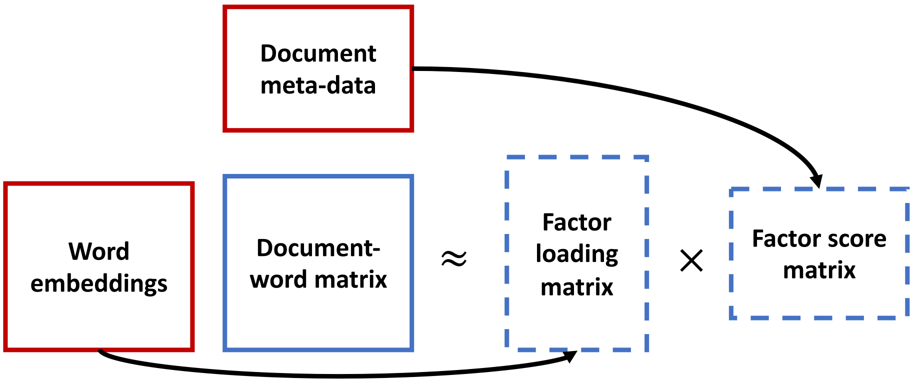


Figure 5.1: Model framework of Zhao et al. [2017c, 2018a, 2017b]. The blue rectangles with solid lines and dash lines are the data matrix (the document-word matrix containing word occurrences of the documents) and the latent matrices (the topic-word and doc-topic distributions), respectively. The red rectangles are the matrices of meta-data (the document meta-data and word embeddings).

a small subset of words in the vocabulary. Therefore, focused topic models [Williamson et al., 2010, Archambeau et al., 2015] have been developed to allow topics to focus on the words that actually explain them. In the paper of [Zhao et al., 2017b], a focused topic model is proposed, where the focusing of topics are informed by external word embeddings. To achieve this, topic embeddings are introduced to interact with pre-trained word embeddings, which determines whether a topic should focus on a word. In this work, besides better performance, the contribution of this work includes the idea of encoding the semantics of topics into the same space of word embeddings, which is an elegant solution of capturing out-of-vocabulary words. The details of this research are in the following attached paper:

- **H. Zhao**, L. Du, W. Buntine, "A Word Embeddings Informed Focused Topic Model", in *Asian Conference on Machine Learning* (**ACML**) 2017.

Incorporating word embeddings in topic modelling is a relatively new direction in topic modelling. Although the model in Zhao et al. [2017c, 2018a] is a relatively efficient approach, it requires binarisation, which may not be a good choice, especially for word embeddings. The model introduced in [Zhao et al., 2017b] is able to work with real-valued word embeddings, but its inference speed can be slow for large collections of documents and word embeddings of large dimensions. In terms of future research, a possible direction is to scale up those models with more scale inference algorithms such as stochastic variational inference and SGMCMC.

The framework of the above models is in Figure 5.1, which can be viewed as the extensions of the basic framework of BLFMs shown in Figure 2.1 in Section 2.2.6 of Chapter 2. Specifically, the proposed models factorise the word occurrences matrix of a collection of documents into the factor loading matrix (the topic-word distributions) and factor score matrix (the document-topic distributions), where the former and latter are informed by the word embeddings and document meta-data, respectively.

The code of this research in this chapter is released at `https://github.com/ethanhezhao/MetaLDA`.

2017 IEEE International Conference on Data Mining

MetaLDA: a Topic Model that Efficiently Incorporates Meta information

He Zhao*, Lan Du*, Wray Buntine* and Gang Liu†
*Faculty of Information Technology
Monash University, Melbourne VIC, Australia
Email: {he.zhao, lan.du, wray.buntine}@monash.edu
†College of Computer Science and Technology
Harbin Engineering University, Harbin, China
Email: liugang@hrbeu.edu.cn

***Abstract*—Besides the text content, documents and their associated words usually come with rich sets of meta information, such as categories of documents and semantic/syntactic features of words, like those encoded in word embeddings. Incorporating such meta information directly into the generative process of topic models can improve modelling accuracy and topic quality, especially in the case where the word-occurrence information in the training data is insufficient. In this paper, we present a topic model, called MetaLDA, which is able to leverage either document or word meta information, or both of them jointly. With two data argumentation techniques, we can derive an efficient Gibbs sampling algorithm, which benefits from the fully local conjugacy of the model. Moreover, the algorithm is favoured by the sparsity of the meta information. Extensive experiments on several real world datasets demonstrate that our model achieves comparable or improved performance in terms of both perplexity and topic quality, particularly in handling sparse texts. In addition, compared with other models using meta information, our model runs significantly faster.**

***Keywords*-topic models; meta information; short texts;**

I. Introduction

With the rapid growth of the internet, huge amounts of text data are generated in social networks, online shopping and news websites, etc. These data create demand for powerful and efficient text analysis techniques. Probabilistic topic models such as Latent Dirichlet Allocation (LDA) [1] are popular approaches for this task, by discovering latent topics from text collections. Many conventional topic models discover topics purely based on the word-occurrences, ignoring the *meta information* (a.k.a., *side information*) associated with the content. In contrast, when we humans read text it is natural to leverage meta information to improve our comprehension, which includes categories, authors, timestamps, the semantic meanings of the words, etc. Therefore, topic models capable of using meta information should yield improved modelling accuracy and topic quality.

In practice, various kinds of meta information are available at the document level and the word level in many corpora. At the document level, labels of documents can be used to guide topic learning so that more meaningful topics can be discovered. Moreover, it is highly likely that documents with common labels discuss similar topics, which could further result in similar topic distributions. For example, if we use authors as labels for scientific papers, the topics of the papers published by the same researcher can be closely related.

At the word level, different semantic/syntactic features are also accessible. For example, there are features regarding word relationships, such as synonyms obtained from WordNet [2], word co-occurrence patterns obtained from a large corpus, and linked concepts from knowledge graphs. It is preferable that words having similar meaning but different morphological forms, like “dog” and “puppy”, are assigned to the same topic, even if they barely co-occur in the modelled corpus. Recently, word embeddings generated by GloVe [3] and word2vec [4], have attracted a lot of attention in natural language processing and related fields. It has been shown that the word embeddings can capture both the semantic and syntactic features of words so that similar words are close to each other in the embedding space. It seems reasonable to expect that these word embedding will improve topic modelling [5], [6].

Conventional topic models can suffer from a large performance degradation over short texts (e.g., tweets and news headlines) because of insufficient word co-occurrence information. In such cases, meta information of documents and words can play an important role in analysing short texts by compensating the lost information in word co-occurrences. At the document level, for example, tweets are usually associated with hashtags, users, locations, and timestamps, which can be used to alleviate the data sparsity problem. At the word level, word semantic similarity and embeddings obtained or trained on large external corpus (e.g., Google News or Wikipedia) have been proven useful in learning meaningful topics from short texts [7], [8].

The benefit of using document and word meta information separately is shown in several models such as [9], [10], [6]. However, in existing models this is usually not efficient enough due to non-conjugacy and/or complex model structures. Moreover, only one kind of meta information (either at document level or at word level) is used in most

2374-8486/17 $31.00 © 2017 IEEE
DOI 10.1109/ICDM.2017.73

existing models. In this paper, we propose MetaLDA[1], a topic model that can effectively and efficiently leverage arbitrary document and word meta information encoded in binary form. Specifically, the labels of a document in MetaLDA are incorporated in the prior of the per-document topic distributions. If two documents have similar labels, their topic distributions should be generated with similar Dirichlet priors. Analogously, at the word level, the features of a word are incorporated in the prior of the per-topic word distributions, which encourages words with similar features to have similar weights across topics. Therefore, both document and word meta information, if and when they are available, can be flexibly and simultaneously incorporated using MetaLDA. MetaLDA has the following key properties:

1) MetaLDA jointly incorporates various kinds of document and word meta information for both regular and short texts, yielding better modelling accuracy and topic quality.
2) With the data augmentation techniques, the inference of MetaLDA can be done by an efficient and closed-form Gibbs sampling algorithm that benefits from the full local conjugacy of the model.
3) The simple structure of incorporating meta information and the efficient inference algorithm give MetaLDA advantage in terms of running speed over other models with meta information.

We conduct extensive experiments with several real datasets including regular and short texts in various domains. The experimental results demonstrate that MetaLDA achieves improved performance in terms of perplexity, topic coherence, and running time.

II. Related Work

In this section, we review three lines of related work: models with document meta information, models with word meta information, and models for short texts.

At the document level, Supervised LDA (sLDA) [11] models document labels by learning a generalised linear model with an appropriate link function and exponential family dispersion function. But the restriction for sLDA is that one document can only have one label. Labelled LDA (LLDA) [12] assumes that each label has a corresponding topic and a document is generated by a mixture of the topics. Although multiple labels are allowed, LLDA requires that the number of topics must equal to the number of labels, i.e., exactly one topic per label. As an extension to LLDA, Partially Labelled LDA (PLLDA) [10] relaxes this requirement by assigning multiple topics to a label. The Dirichlet Multinomial Regression (DMR) model [9] incorporates document labels on the prior of the topic distributions like our MetaLDA but with the logistic-normal transformation. As full conjugacy does not exist in DMR, a part of the inference has to be done by numerical optimisation, which is slow for large sets of labels and topics. Similarly, in the Hierarchical Dirichlet Scaling Process (HDSP) [13], conjugacy is broken as well since the topic distributions have to be renormalised. [14] introduces a Poisson factorisation model with hierarchical document labels. But the techniques cannot be applied to regular topic models as the topic proportion vectors are also unnormalised.

Recently, there is growing interest in incorporating word features in topic models. For example, DF-LDA [15] incorporates word must-links and cannot-links using a Dirichlet forest prior in LDA; MRF-LDA [16] encodes word semantic similarity in LDA with a Markov random field; WF-LDA [17] extends LDA to model word features with the logistic-normal transform; LF-LDA [6] integrates word embeddings into LDA by replacing the topic-word Dirichlet multinomial component with a mixture of a Dirichlet multinomial component and a word embedding component; Instead of generating word types (tokens), Gaussian LDA (GLDA) [5] directly generates word embeddings with the Gaussian distribution. Despite the exciting applications of the above models, their inference is usually less efficient due to the non-conjugacy and/or complicated model structures.

Analysis of short text with topic models has been an active area with the development of social networks. Generally, there are two ways to deal with the sparsity problem in short texts, either using the intrinsic properties of short texts or leveraging meta information. For the first way, one popular approach is to aggregate short texts into pseudo-documents, for example, [18] introduces a model that aggregates tweets containing the same word; Recently, PTM [19] aggregates short texts into latent pseudo documents. Another approach is to assume one topic per short document, known as mixture of unigrams or Dirichlet Multinomial Mixture (DMM) such as [20], [7]. For the second way, document meta information can be used to aggregate short texts, for example, [18] aggregates tweets by the corresponding authors and [21] shows that aggregating tweets by their hashtags yields superior performance over other aggregation methods. One closely related work to ours is the models that use word features for short texts. For example, [7] introduces an extension of GLDA on short texts which samples an indicator variable that chooses to generate either the type of a word or the embedding of a word and GPU-DMM [8] extends DMM with word semantic similarity obtained from embeddings for short texts. Although with improved performance there still exists challenges for existing models: (1) for aggregation-based models, it is usually hard to choose which meta information to use for aggregation; (2) the "single topic" assumption makes DMM models lose the flexibility to capture different topic ingredients of a document; and (3) the incorporation of meta information in the existing models is usually less efficient.

To our knowledge, the attempts that jointly leverage

[1] Code at https://github.com/ethanhezhao/MetaLDA/

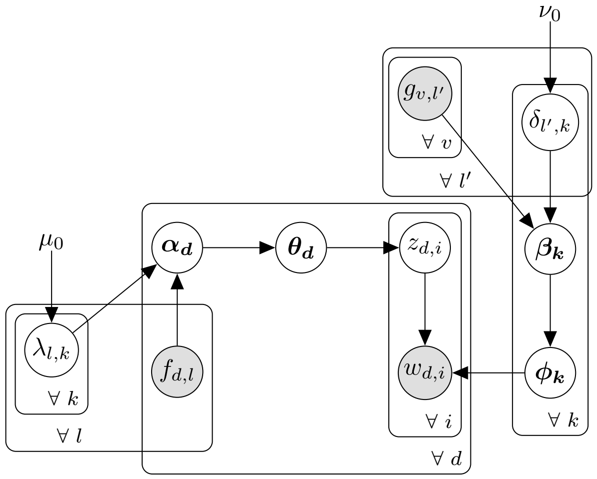


Figure 1: The graphical model of MetaLDA

document and word meta information are relatively rare. For example, meta information can be incorporated by first-order logic in Logit-LDA [22] and score functions in SC-LDA [23]. However, the first-order logic and score functions need to be defined for different kinds of meta information and the definition can be infeasible for incorporating both document and word meta information simultaneously.

III. The MetaLDA Model

Given a corpus, LDA uses the same Dirichlet prior for all the per-document topic distributions and the same prior for all the per-topic word distributions [24]. While in MetaLDA, each document has a specific Dirichlet prior on its topic distribution, which is computed from the meta information of the document, and the parameters of the prior are estimated during training. Similarly, each topic has a specific Dirichlet prior computed from the word meta information. Here we elaborate our MetaLDA, in particular on how the meta information is incorporated. Hereafter, we will use labels as document meta information, unless otherwise stated.

Given a collection of D documents $\mathcal{D}$, MetaLDA generates document $d \in \{1, \cdots, D\}$ with a mixture of K topics and each topic $k \in \{1, \cdots, K\}$ is a distribution over the vocabulary with V tokens, denoted by $\boldsymbol{\phi}_k \in \mathbb{R}_+^V$. For document d with N_d words, to generate the i^{th} ($i \in \{1, \cdots, N_d\}$) word $w_{d,i}$, we first sample a topic $z_{d,i} \in \{1, \cdots, K\}$ from the document's topic distribution $\boldsymbol{\theta_d} \in \mathbb{R}_+^K$, and then sample $w_{d,i}$ from $\boldsymbol{\phi}_{z_{d,i}}$. Assume the labels of document d are encoded in a binary vector $\boldsymbol{f_d} \in \{0,1\}^{L_{doc}}$ where L_{doc} is the total number of unique labels. $f_{d,l} = 1$ indicates label l is active in document d and vice versa. Similarly, the L_{word} features of token v are stored in a binary vector $\boldsymbol{g}_v \in \{0,1\}^{L_{word}}$. Therefore, the document and word meta information associated with $\mathcal{D}$ are stored in the matrix $\mathbf{F} \in \{0,1\}^{D \times L_{doc}}$ and $\mathbf{G} \in \{0,1\}^{V \times L_{word}}$ respectively. Although MetaLDA incorporates binary features, categorical features and real-valued features can be converted into binary values with proper transformations such as discretisation and binarisation.

Fig. 1 shows the graphical model of MetaLDA and the generative process is as following:

1) For each topic k:
   a) For each doc-label l: Draw $\lambda_{l,k} \sim \text{Ga}(\mu_0, \mu_0)$
   b) For each word-feat l': Draw $\delta_{l',k} \sim \text{Ga}(\nu_0, \nu_0)$
   c) For each token v: Compute $\beta_{k,v} = \prod_{l'=1}^{L_{word}} \delta_{l',k}^{g_{v,l'}}$
   d) Draw $\boldsymbol{\phi}_k \sim \text{Dir}_V(\boldsymbol{\beta}_k)$
2) For each document d:
   a) For each topic k: Compute $\alpha_{d,k} = \prod_{l=1}^{L_{doc}} \lambda_{l,k}^{f_{d,l}}$
   b) Draw $\boldsymbol{\theta}_d \sim \text{Dir}_K(\boldsymbol{\alpha}_d)$
   c) For each word in document d:
      i) Draw topic $z_{d,i} \sim \text{Cat}_K(\boldsymbol{\theta}_d)$
      ii) Draw word $w_{d,i} \sim \text{Cat}_V(\boldsymbol{\phi}_{z_{d,i}})$

where $\text{Ga}(\cdot,\cdot)$, $\text{Dir}(\cdot)$, $\text{Cat}(\cdot)$ are the gamma distribution, the Dirichlet distribution, and the categorical distribution respectively. K, μ_0, and ν_0 are the hyper-parameters.

To incorporate document labels, MetaLDA learns a specific Dirichlet prior over the topics for each document by using the label information. Specifically, the information of document d's labels is incorporated in $\boldsymbol{\alpha}_d$, the parameter of Dirichlet prior on $\boldsymbol{\theta}_d$. As shown in Step 2a, $\alpha_{d,k}$ is computed as a log linear combination of the labels $f_{d,l}$. Since $f_{d,l}$ is binary, $\alpha_{d,k}$ is indeed the multiplication of $\lambda_{l,k}$ over all the active labels of document d, i.e., $\{l \mid f_{d,l} = 1\}$. Drawn from the gamma distribution with mean 1, $\lambda_{l,k}$ controls the impact of label l on topic k. If label l has no or less impact on topic k, $\lambda_{l,k}$ is expected to be 1 or close to 1, and then $\lambda_{l,k}$ will have no or little influence on $\alpha_{d,k}$ and vice versa. The hyper-parameter μ_0 controls the variation of $\lambda_{l,k}$. The incorporation of word features is analogous but in the parameter of the Dirichlet prior on the per-topic word distributions as shown in Step 1c.

The intuition of our way of incorporating meta information is: At the document level, if two documents have more labels in common, their Dirichlet parameter $\boldsymbol{\alpha}_d$ will be more similar, resulting in more similar topic distributions $\boldsymbol{\theta}_d$; At the word level, if two words have similar features, their $\beta_{k,v}$ in topic k will be similar and then we can expect that their $\phi_{k,v}$ could be more or less the same. Finally, the two words will have similar probabilities of showing up in topic k. In other words, if a topic "prefers" a certain word, we expect that it will also prefer other words with similar features to that word. Moreover, at both the document and the word level, different labels/features may have different impact on the topics (λ/δ), which is automatically learnt in MetaLDA.

IV. Inference

Unlike most existing methods, our way of incorporating the meta information facilitates the derivation of an efficient

Gibbs sampling algorithm. With two data augmentation techniques (i.e., the introduction of auxiliary variables), MetaLDA admits the local conjugacy and a close-form Gibbs sampling algorithm can be derived. Note that MetaLDA incorporates the meta information on the Dirichlet priors, so we can still use LDA's collapsed Gibbs sampling algorithm for the topic assignment $z_{d,i}$. Moreover, Step 2a and 1c show that one only needs to consider the non-zero entries of $\mathbf{F}$ and $\mathbf{G}$ in computing the full conditionals, which further reduces the inference complexity.

Similar to LDA, the complete model likelihood (i.e., joint distribution) of MetaLDA is:

$$\prod_{k=1}^{K}\prod_{v=1}^{V}\phi_{k,v}^{n_{k,v}} \cdot \prod_{d=1}^{D}\prod_{k=1}^{K}\theta_{d,k}^{m_{d,k}} \tag{1}$$

where $n_{k,v} = \sum_{d}^{D}\sum_{i=1}^{N_d}\mathbf{1}_{(w_{d,i}=v, z_{d,i}=k)}$, $m_{d,k} = \sum_{i=1}^{N_d}\mathbf{1}_{(z_{d,i}=k)}$, and $\mathbf{1}_{(\cdot)}$ is the indicator function.

A. Sampling $\lambda_{l,k}$:

To sample $\lambda_{l,k}$, we first marginalise out $\theta_{d,k}$ in the right part of Eq. (1) with the Dirichlet multinomial conjugacy:

$$\prod_{d=1}^{D}\underbrace{\frac{\Gamma(\alpha_{d,\cdot})}{\Gamma(\alpha_{d,\cdot}+m_{d,\cdot})}}_{\text{Gamma ratio 1}}\prod_{k=1}^{K}\underbrace{\frac{\Gamma(\alpha_{d,k}+m_{d,k})}{\Gamma(\alpha_{d,k})}}_{\text{Gamma ratio 2}} \tag{2}$$

where $\alpha_{d,\cdot} = \sum_{k=1}^{K}\alpha_{d,k}$, $m_{d,\cdot} = \sum_{k=1}^{K}m_{d,k}$, and $\Gamma(\cdot)$ is the gamma function. Gamma ratio 1 in Eq. (2) can be augmented with a set of Beta random variables $q_{1:D}$ as:

$$\underbrace{\frac{\Gamma(\alpha_{d,\cdot})}{\Gamma(\alpha_{d,\cdot}+m_{d,\cdot})}}_{\text{Gamma ratio 1}} \propto \int_{q_d} q_d^{\alpha_{d,\cdot}-1}(1-q_d)^{m_{d,\cdot}-1} \tag{3}$$

where for each document d, $q_d \sim \text{Beta}(\alpha_{d,\cdot}, m_{d,\cdot})$. Given a set of $q_{1:D}$ for all the documents, Gamma ratio 1 can be approximated by the product of $q_{1:D}$, i.e., $\prod_{d=1}^{D} q_d^{\alpha_{d,\cdot}}$.

Gamma ratio 2 in Eq. (2) is the Pochhammer symbol for a rising factorial, which can be augmented with an auxiliary variable $t_{d,k}$ [25], [26], [27], [28] as follows:

$$\underbrace{\frac{\Gamma(\alpha_{d,k}+m_{d,k})}{\Gamma(\alpha_{d,k})}}_{\text{Gamma ratio 2}} = \sum_{t_{d,k}=0}^{m_{d,k}} S_{t_{d,k}}^{m_{d,k}}\alpha_{d,k}^{t_{d,k}} \tag{4}$$

where S_t^m indicates an unsigned Stirling number of the first kind. Gamma ratio 2 is a normalising constant for the probability of the number of tables in the Chinese Restaurant Process (CRP) [29], $t_{d,k}$ can be sampled by a CRP with $\alpha_{d,k}$ as the concentration and $m_{d,k}$ as the number of customers:

$$t_{d,k} = \sum_{i=1}^{m_{d,k}} \text{Bern}\left(\frac{\alpha_{d,k}}{\alpha_{d,k}+i}\right) \tag{5}$$

where Bern$(\cdot)$ samples from the Bernoulli distribution. The complexity of sampling $t_{d,k}$ by Eq. (5) is $\mathcal{O}(m_{d,k})$. For large $m_{d,k}$, as the standard deviation of $t_{d,k}$ is $\mathcal{O}(\sqrt{\log m_{d,k}})$ [29], one can sample $t_{d,k}$ in a small window around the current value in complexity $\mathcal{O}(\sqrt{\log m_{d,k}})$.

By ignoring the terms unrelated to α, the augmentation of Eq. (4) can be simplified to a single term $\alpha_{d,k}^{t_{d,k}}$. With auxiliary variables now introduced, we simplify Eq. (2) to:

$$\prod_{d=1}^{D}\prod_{k=1}^{K} q_d^{\alpha_{d,k}}\alpha_{d,k}^{t_{d,k}} \tag{6}$$

Replacing $\alpha_{d,k}$ with $\lambda_{l,k}$, we can get:

$$\prod_{d=1}^{D}\prod_{k=1}^{K} e^{-\alpha_{d,k}\log\frac{1}{q_d}} \cdot \prod_{l=1}^{L_{doc}}\prod_{k=1}^{K}\lambda_{l,k}^{\sum_{d=1}^{D} f_{d,l}t_{d,k}}$$

Recall that all the document labels are binary and $\lambda_{l,k}$ is involved in computing $\alpha_{d,k}$ iff $f_{d,l} = 1$. Extracting all the terms related to $\lambda_{l,k}$ in Eq. (7), we get the marginal posterior of $\lambda_{l,k}$:

$$e^{-\lambda_{l,k}\sum_{d=1:f_{d,l}=1}^{D}\log\frac{1}{q_d}\cdot\frac{\alpha_{d,k}}{\lambda_{l,k}}}\lambda_{l,k}^{\sum_{d=1}^{D} f_{d,l}t_{d,k}}$$

where $\frac{\alpha_{d,k}}{\lambda_{l,k}}$ is the value of $\alpha_{d,k}$ with $\lambda_{l,k}$ removed when $f_{d,l} = 1$. With the data augmentation techniques, the posterior is transformed into a form that is conjugate to the gamma prior of $\lambda_{l,k}$. Therefore, it is straightforward to yield the following sampling strategy for $\lambda_{l,k}$:

$$\lambda_{l,k} \sim \text{Ga}(\mu', 1/\mu'') \tag{7}$$

$$\mu' = \mu_0 + \sum_{d=1:f_{d,l}=1}^{D} t_{d,k} \tag{8}$$

$$\mu'' = 1/\mu_0 - \sum_{d=1:f_{d,l}=1}^{D}\frac{\alpha_{d,k}}{\lambda_{l,k}}\log q_d \tag{9}$$

We can compute and cache the value of $\alpha_{d,k}$ first. After $\lambda_{l,k}$ is sampled, $\alpha_{d,k}$ can be updated by:

$$\alpha_{d,k} \leftarrow \frac{\alpha_{d,k}\lambda'_{l,k}}{\lambda_{l,k}} \;\forall\; 1 \le d \le D : f_{d,l} = 1 \tag{10}$$

where $\lambda'_{i,k}$ is the newly-sampled value of $\lambda_{i,k}$.

To sample/compute Eqs. (7)-(10), one only iterates over the documents where label l is active (i.e., $f_{d,l} = 1$). Thus, the sampling for all λ takes $\mathcal{O}(D'KL_{doc})$ where D' is the average number of documents where a label is active (i.e., the column-wise sparsity of $\mathbf{F}$). It is usually that $D' \ll D$ because if a label exists in nearly all the documents, it provides little discriminative information. This demonstrates how the sparsity of document meta information is leveraged. Moreover, sampling all the tables t takes $\mathcal{O}(\tilde{N})$ ($\tilde{N}$ is the total number of words in $\mathcal{D}$) which can be accelerated with the window sampling technique explained above.

B. Sampling $\delta_{l',k}$:

Since the derivation of sampling $\delta_{l',k}$ is analogous to $\lambda_{l,k}$, we directly give the sampling formulas:

$$\delta_{l',k} \sim \mathrm{Ga}(\nu', 1/\nu'') \tag{11}$$

$$\nu' = \nu_0 + \sum_{v=1:g_{v,l'}=1}^{V} t'_{k,v} \tag{12}$$

$$\nu'' = 1/\nu_0 - \log q'_k \sum_{v=1:g_{v,l'}=1}^{V} \frac{\beta_{k,v}}{\delta_{l',k}} \tag{13}$$

where the two auxiliary variables can be sampled by: $q'_k \sim \mathrm{Beta}(\beta_{k,\cdot}, n_{k,\cdot})$ and $t'_{k,v} \sim \mathrm{CRP}(\beta_{k,v}, n_{k,v})$. Similarly, sampling all δ takes $\mathcal{O}(V'KL_{word})$ where V' is the average number of tokens where a feature is active (i.e., the column-wise sparsity of $\mathbf{G}$ and usually $V' \ll V$) and sampling all the tables t' takes $\mathcal{O}(\tilde{N})$.

C. Sampling topic $z_{d,i}$:

Given $\boldsymbol{\alpha_d}$ and $\boldsymbol{\beta_k}$, the collapsed Gibbs sampling of a new topic for a word $w_{d,i} = v$ in MetaLDA is:

$$\Pr(z_{d,i} = k) \propto (\alpha_{d,k} + m_{d,k}) \frac{\beta_{k,v} + n_{k,v}}{\beta_{k,\cdot} + n_{k,\cdot}} \tag{14}$$

which is exactly the same to LDA.

V. Experiments

In this section, we evaluate the proposed MetaLDA against several recent advances that also incorporate meta information on 6 real datasets including both regular and short texts. The goal of the experimental work is to evaluate the effectiveness and efficiency of MetaLDA's incorporation of document and word meta information both separately and jointly compared with other methods. We report the performance in terms of perplexity, topic coherence, and running time per iteration.

A. Datasets

In the experiments, three regular text datasets and three short text datasets were used:

- **Reuters** is widely used corpus extracted from the Reuters-21578 dataset where documents without any labels are removed[2]. There are 11,367 documents and 120 labels. Each document is associated with multiple labels. The vocabulary size is 8,817 and the average document length is 73.
- **20NG**, 20 Newsgroup, a widely used dataset consists of 18,846 news articles with 20 categories. The vocabulary size is 22,636 and the average document length is 108.
- **NYT**, New York Times is extracted from the documents in the category "Top/News/Health" in the New York Times Annotated Corpus[3]. There are 52,521 documents and 545 unique labels. Each document is with multiple labels. The vocabulary contains 21,421 tokens and there are 442 words in a document on average.
- **WS**, Web Snippet, used in [8], contains 12,237 web search snippets and each snippet belongs to one of 8 categories. The vocabulary contains 10,052 tokens and there are 15 words in one snippet on average.
- **TMN**, Tag My News, used in [6], consists of 32,597 English RSS news snippets from Tag My News. With a title and a short description, each snippet belongs to one of 7 categories. There are 13,370 tokens in the vocabulary and the average length of a snippet is 18.
- **AN**, ABC News, is a collection of 12,495 short news descriptions and each one is in multiple of 194 categories. There are 4,255 tokens in the vocabulary and the average length of a description is 13.

All the datasets were tokenised by Mallet[4] and we removed the words that exist in less than 5 documents and more than 95% documents.

B. Meta Information Settings

Document labels and word features. At the document level, the labels associated with documents in each dataset were used as the meta information. At the word level, we used a set of 100-dimensional binarised word embeddings as word features[2], which were obtained from the 50-dimensional GloVe word embeddings pre-trained on Wikipedia[5]. To binarise word embeddings, we first adopted the following method similar to [30]:

$$g'_{v,j} = \begin{cases} 1, & \text{if } g''_{v,j} > \mathrm{Mean}_{+}(\boldsymbol{g}''_v) \\ -1, & \text{if } g''_{v,j} < \mathrm{Mean}_{-}(\boldsymbol{g}''_v) \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

where $\boldsymbol{g}''_v$ is the original embedding vector for word v, $g'_{v,j}$ is the binarised value for j^{th} element of $\boldsymbol{g''_v}$, and $\mathrm{Mean}_{+}(\cdot)$ and $\mathrm{Mean}_{-}(\cdot)$ are the average value of all the positive elements and negative elements respectively. The insight is that we only consider features with strong opinions (i.e., large positive or negative value) on each dimension. To transform $g' \in \{-1, 1\}$ to the final $g \in \{0, 1\}$, we use two binary bits to encode one dimension of $g'_{v,j}$: the first bit is on if $g'_{v,j} = 1$ and the second is on if $g'_{v,j} = -1$. Besides, MetaLDA can work with other word features such as semantic similarity as well.

Default feature. Besides the labels/features associated with the datasets, a default label/feature for each document/word is introduced in MetaLDA, which is always equal to 1. The default can be interpreted as the bias term in α/β, which captures the information unrelated to the

[2] MetaLDA is able to handle documents/words without labels/features. But for fair comparison with other models, we removed the documents without labels and words without features.

[3] https://catalog.ldc.upenn.edu/ldc2008t19

[4] http://mallet.cs.umass.edu

[5] https://nlp.stanford.edu/projects/glove/

Table I: MetaLDA and its variants.

	Compute α with	Compute β with
MetaLDA	Document labels	Word features
MetaLDA-dl-def	Document labels	Default feature
MetaLDA-dl-0.01	Document labels	Symmetric 0.01 (fixed)
MetaLDA-def-wf	Default label	Word features
MetaLDA-0.1-wf	Symmetric 0.1 (fixed)	Word features
MetaLDA-def-def	Default label	Default feature

labels/features. While there are no document labels or word features, with the default, MetaLDA is equivalent in model to asymmetric-asymmetric LDA of [24].

C. Compared Models and Parameter Settings

We evaluate the performance of the following models:

- **MetaLDA** and its variants: the proposed model and its variants. Here we use MetaLDA to indicate the model considering both document labels and word features. Several variants of MetaLDA with document labels and word features separately were also studied, which are shown in Table I. These variants differ in the method of estimating $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. All the models listed in Table I were implemented on top of Mallet. The hyper-parameters μ_0 and ν_0 were set to 1.0.
- **LDA** [1]: the baseline model. The Mallet implementation of SparseLDA [31] is used.
- **LLDA**, Labelled LDA [12] and **PLLDA**, Partially Labelled LDA [10]: two models that make use of multiple document labels. The original implementation[6] is used.
- **DMR**, LDA with Dirichlet Multinomial Regression [9]: a model that can use multiple document labels. The Mallet implementation of DMR based on SparseLDA was used. Following Mallet, we set the mean of λ to 0.0 and set the variances of λ for the default label and the document labels to 100.0 and 1.0 respectively.
- **WF-LDA**, Word Feature LDA [17]: a model with word features. We implemented it on top of Mallet and used the default settings in Mallet for the optimisation.
- **LF-LDA**, Latent Feature LDA [6]: a model that incorporates word embeddings. The original implementation[7] was used. Following the paper, we used 1500 and 500 MCMC iterations for initialisation and sampling respectively and set λ to 0.6, and used the original 50-dimensional GloVe word embeddings as word features.
- **GPU-DMM**, Generalized Pólya Urn DMM [8]: a model that incorporates word semantic similarity. The original implementation[8] was used. The word similarity was generated from the distances of the word embeddings. Following the paper, we set the hyper-parameters μ and ϵ to 0.1 and 0.7 respectively, and the symmetric document Dirichlet prior to $50/K$.
- **PTM**, Pseudo document based Topic Model [19]: a model for short text analysis. The original implementation[9] was used. Following the paper, we set the number of pseudo documents to 1000 and λ to 0.1.

All the models, except where noted, the symmetric parameters of the document and the topic Dirichlet priors were set to 0.1 and 0.01 respectively, and 2000 MCMC iterations are used to train the models.

D. Perplexity Evaluation

Perplexity is a measure that is widely used [24] to evaluate the modelling accuracy of topic models. The lower the score, the higher the modelling accuracy. To compute perplexity, we randomly selected some documents in a dataset as the training set and the remaining as the test set. We first trained a topic model on the training set to get the word distributions of each topic k ($\boldsymbol{\phi}_k^{train}$). Each test document d was split into two halves containing every first and every second words respectively. We then fixed the topics and trained the models on the first half to get the topic proportions ($\boldsymbol{\theta}_d^{test}$) of test document d and compute perplexity for predicting the second half. In regard to MetaLDA, we fixed the matrices $\boldsymbol{\Phi}^{train}$ and $\boldsymbol{\Lambda}^{train}$ output from the training procedure. On the first half of test document d, we computed the Dirichlet prior $\boldsymbol{\alpha}_d^{test}$ with $\boldsymbol{\Lambda}^{train}$ and the labels $\boldsymbol{f}_d^{test}$ of test document d (See Step 2a), and then point-estimated $\boldsymbol{\theta}_d^{test}$. We ran all the models 5 times with different random number seeds and report the average scores and the standard deviations.

In testing, we may encounter words that never occur in the training documents (a.k.a., unseen words or out-of-vocabulary words). There are two strategies for handling unseen words for calculating perplexity on test documents: ignoring them or keeping them in computing the perplexity. Here we investigate both strategies:

1) Perplexity Computed without Unseen Words: In this experiment, the perplexity is computed only on the words that appear in the training vocabulary. Here we used 80% documents in each dataset as the training set and the remaining 20% as the test set.

Tables II and III show[10]: the average perplexity scores with standard deviations for all the models. Note that: (1) The scores on AN with 150 and 200 topics are not reported due to overfitting observed in all the compared models. (2) Given the size of NYT, the scores of 200 and 500 topics are reported. (3) The number of latent topics in LLDA must equal to the number of document labels. (4) For PLLDA, we varied the number of topics per label from 5 to 50 (2 and 5 topics on NYT). The number of topics in PPLDA is the product of the numbers of labels and topics per label.

[6] https://nlp.stanford.edu/software/tmt/tmt-0.4/

[7] https://github.com/datquocnguyen/LFTM

[8] https://github.com/NobodyWHU/GPUDMM

[9] http://ipv6.nlsde.buaa.edu.cn/zuoyuan/

[10] For GPU-DMM and PTM, perplexity is not evaluated because the inference code for unseen documents is not public available. The random number seeds used in the code of LLDA and PLLDA are pre-fixed in the package. So the standard deviations of the two models are not reported.

Table II: Perplexity comparison on the regular text datasets. The best results are highlighted in boldface.

	Dataset	Reuters				20NG				NYT	
	#Topics	*50*	*100*	*150*	*200*	*50*	*100*	*150*	*200*	*200*	*500*
No meta info	LDA	677±1	634±2	629±1	631±1	2147±7	1930±7	1820±5	1762±3	2293±8	2154±4
	MetaLDA-def-def	648±3	592±2	559±1	540±1	2093±6	1843±7	1708±5	1626±4	2258±9	2079±8
Doc labels	DMR	640±1	577±1	544±2	526±2	2080±8	1811±8	1670±4	1578±1	**2231**±13	**2013**±6
	MetaLDA-dl-0.01	649±2	582±2	551±3	530±2	2067±9	1821±7	1680±5	1590±1	**2219**±4	**2018**±4
	MetaLDA-dl-def	642±3	576±3	543±1	526±1	2050±4	1804±6	1675±8	1589±2	**2230**±3	**2022**±5
Word features	LF-LDA	841±4	787±4	772±3	771±4	2855±21	2576±3	2433±7	2326±8	2831±2	2700±5
	WF-LDA	659±2	616±2	615±1	613±1	2089±7	1875±2	1784±2	1727±3	2287±6	2134±6
	MetaLDA-0.1-wf	659±3	621±1	619±1	623±1	2098±7	1887±8	1796±8	1744±4	2283±4	2143±2
	MetaLDA-def-wf	643±2	582±4	552±3	535±1	2068±6	1819±1	1685±7	1600±3	2260±7	2095±6
Doc labels & word features ⟶	MetaLDA	**633**±2	**568**±2	**536**±2	**517**±1	**2025**±12	**1781**±8	**1640**±5	**1551**±6	**2217**±6	**2020**±6

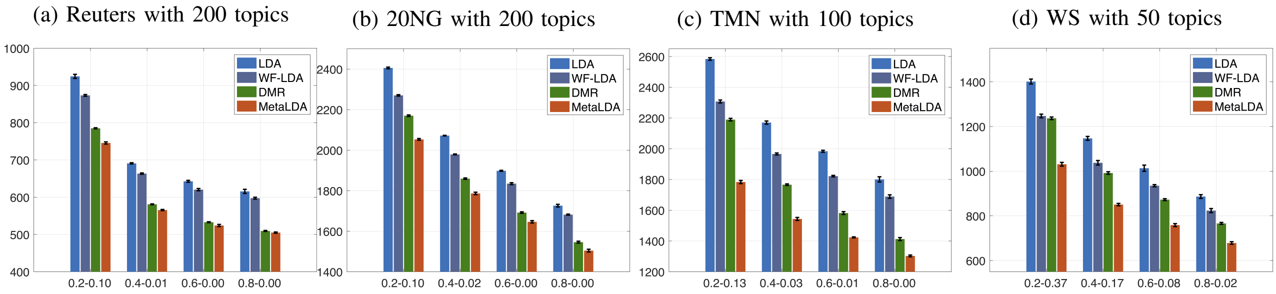
	Dataset	Reuters				20NG				NYT	
	#Topics per label	*5*	*10*	*20*	*50*	*5*	*10*	*20*	*50*	*2*	*5*
Doc labels	PLLDA	714	708	733	829	1997	1786	1605	**1482**	2839	2846
	LLDA	834				2607				2948	

Table III: Perplexity comparison without unseen words on the short text datasets. The best results are highlighted in boldface.

	Dataset	WS				TMN				AN	
	#Topics	*50*	*100*	*150*	*200*	*50*	*100*	*150*	*200*	*50*	*100*
No meta info	LDA	961±6	878±8	869±6	888±5	1969±14	1873±6	1881±9	1916±4	406±14	422±12
	MetaLDA-def-def	884±10	733±6	671±6	625±6	1800±11	1578±19	1469±4	1422±6	352±16	336±11
Doc labels	DMR	845±7	683±4	607±1	562±2	1750±8	1506±3	1391±7	1323±5	326±6	**290**±5
	MetaLDA-dl-0.01	840±7	693±6	618±3	588±4	1767±11	1528±10	1416±7	1345±13	321±13	303±8
	MetaLDA-dl-def	832±4	679±5	622±7	582±5	1720±7	1505±16	1395±11	1325±12	319±9	293±7
Word features	LF-LDA	1164±6	1039±17	1019±11	992±6	2415±35	2393±11	2371±10	2374±14	482±17	514±19
	WF-LDA	894±6	839±6	827±10	842±4	1853±6	1766±12	1830±60	1854±45	397±5	410±6
	MetaLDA-0.1-wf	889±6	832±3	839±2	853±4	1865±4	1784±2	1799±9	1831±6	388±3	410±8
	MetaLDA-def-wf	830±6	688±8	624±5	584±4	1730±14	1504±3	1402±13	1342±4	346±15	332±8
Doc labels & word features ⟶	MetaLDA	**774**±9	**627**±6	**572**±3	**534**±4	**1657**±4	**1415**±16	**1304**±6	**1235**±6	**314**±9	**293**±9

	Dataset	WS				TMN				AN	
	#Topics per label	*5*	*10*	*20*	*50*	*5*	*10*	*20*	*50*	*5*	*10*
Doc labels	PLLDA	1060	886	735	642	2181	1863	1647	1456	440	525
	LLDA	1543				2958				392	

Figure 2: Perplexity comparison with unseen words in different proportions of the training documents. Each pair of the numbers on the horizontal axis are the proportion of the training documents and the proportion of unseen tokens in the vocabulary of the test documents, respectively. The error bars are the standard deviations over 5 runs.



(a) Reuters with 200 topics

(b) 20NG with 200 topics

(c) TMN with 100 topics

(d) WS with 50 topics

The results show that MetaLDA outperformed all the competitors in terms of perplexity on nearly all the datasets, showing the benefit of using both document and word meta information. Specifically, we have the following remarks:

- By looking at the models using only the document-level meta information, we can see the significant improvement of these models over LDA, which indicates that document labels can play an important role in guiding topic modelling. Although the performance of the two variants of MetaLDA with document labels and DMR is comparable, our models runs much faster than DMR, which will be studied later in Section V-F.
- It is interesting that PLLDA with 50 topics for each label has better perplexity than MetaLDA with 200 topics in the 20NG dataset. With the 20 unique labels, the actual number of topics in PLLDA is 1000. However, if 10 topics for each label in PLLDA are used, which is equivalent to 200 topics in MetaLDA, PLLDA is outperformed by MetaLDA significantly.
- At the word level, MetaLDA-def-wf performed the best

among the models with word features only. Moreover, our model has obvious advantage in running speed (see Table V). Furthermore, comparing MetaLDA-def-wf with MetaLDA-def-def and MetaLDA-0.1-wf with LDA, we can see using the word features indeed improved perplexity.

- The scores show that the improvement gained by MetaLDA over LDA on the short text datasets is larger than that on the regular text datasets. This is as expected because meta information serves as complementary information in MetaLDA and can have more significant impact when the data is sparser.
- It can be observed that models usually gained improved perplexity, if α is sampled/optimised, in line with [24].
- On the AN dataset, there is no statistically significant difference between MetaLDA and DMR. On NYT, a similar trend is observed: the improvement in the models with the document labels over LDA is obvious but not in the models with the word features. Given the number of the document labels (194 of AN and 545 of NYT), it is possible that the document labels already offer enough information and the word embeddings have little contribution in the two datasets.

2) Perplexity Computed with Unseen Words: To test the hypothesis that the incorporation of meta information in MetaLDA can significantly improve the modelling accuracy in the cases where the corpus is sparse, we varied the proportion of documents used in training from 20% to 80% and used the remaining for testing. It is natural that when the proportion is small, the number of unseen words in testing documents will be large. Instead of simply excluding the unseen words in the previous experiments, here we compute the perplexity with unseen words for LDA, DMR, WF-LDA and the proposed MetaLDA. For perplexity calculation, $\phi_{k,v}^{test}$ for each topic k and each token v in the test documents is needed. If v occurs in the training documents, $\phi_{k,v}^{test}$ can be directly obtained. While if v is unseen, $\phi_{k,v}^{unseen}$ can be estimated by the prior: $\frac{\beta_{k,v}^{unseen}}{n_{k,\cdot}^{train}+\beta_{k,\cdot}^{train}+\beta_{k,\cdot}^{unseen}}$. For LDA and DMR which do not use word features, $\beta_{k,v}^{unseen} = \beta_{k,v}^{train}$; For WF-LDA and MetaLDA which are with word features, $\beta_{k,v}^{unseen}$ is computed with the features of the unseen token. Following Step 1c, for MetaLDA, $\beta_{k,v}^{unseen} = \prod_{l'}^{L_{word}} \delta_{l',k}^{g_{v,l}^{unseen}}$.

Figure 2 shows the perplexity scores on Reuters, 20NG, TMN and WS with 200, 200, 100 and 50 topics respectively. MetaLDA outperformed the other models significantly with a lower proportion of training documents and relatively higher proportion of unseen words. The gap between MetaLDA and the other three models increases while the training proportion decreases. It indicates that the meta information helps MetaLDA to achieve better modelling accuracy on predicting unseen words.

E. Topic Coherence Evaluation

We further evaluate the semantic coherence of the words in a topic learnt by LDA, PTM, DMR, LF-LDA, WF-LDA, GPU-DMM and MetaLDA. Here we use the Normalised Pointwise Mutual Information (NPMI) [32], [33] to calculate topic coherence score for topic k with top T words: $\text{NPMI}(k) = \sum_{j=2}^{T} \sum_{i=1}^{j-1} \log \frac{p(w_j,w_i)}{p(w_j)p(w_i)} / - \log p(w_j, w_i)$, where $p(w_i)$ is the probability of word i, and $p(w_i, w_j)$ is the joint probability of words i and j that co-occur together within a sliding window. Those probabilities were computed on an external large corpus, i.e., a 5.48GB Wikipedia dump in our experiments. The NPMI score of each topic in the experiments is calculated with top 10 words ($T = 10$) by the Palmetto package[11]. Again, we report the average scores and the standard deviations over 5 random runs.

It is known that conventional topic models directly applied to short texts suffer from low quality topics, caused by the insufficient word co-occurrence information. Here we study whether or not the meta information helps MetaLDA improve topic quality, compared with other topic models that can also handle short texts. Table IV shows the NPMI scores on the three short text datasets. Higher scores indicate better topic coherence. All the models were trained with 100 topics. Besides the NPMI scores averaged over all the 100 topics, we also show the scores averaged over top 20 topics with highest NPMI, where "rubbish" topics are eliminated, following [23]. It is clear that MetaLDA performed significantly better than all the other models in WS and AN dataset in terms of NPMI, which indicates that MetaLDA can discover more meaningful topics with the document and word meta information. We would like to point out that on the TMN dataset, even though the average score of MetaLDA is still the best, the score of MetaLDA has overlapping with the others' in the standard deviation, which indicates the difference is not statistically significant.

F. Running Time

In this section, we empirically study the efficiency of the models in term of per-iteration running time. The implementation details of our MetaLDA are as follows: (1) The SparseLDA framework [31] reduces the complexity of LDA to be sub-linear by breaking the conditional of LDA into three "buckets", where the "smoothing only" bucket is cached for all the documents and the "document only" bucket is cached for all the tokens in a document. We adopted a similar strategy when implementing MetaLDA. When only the document meta information is used, the Dirichlet parameters α for different documents in MetaLDA are different and asymmetric. Therefore, the "smoothing only" bucket has to be computed for each document, but we can cache it for all the tokens, which still gives us a considerable reduction in computing complexity. However,

[11] http://palmetto.aksw.org

Table IV: Topic coherence (NPMI) on the short text datasets.

		All 100 topics			Top 20 topics		
		WS	TMN	AN	WS	TMN	AN
No meta info	LDA	-0.0030±0.0047	0.0319±0.0032	-0.0636±0.0033	0.1025±0.0067	0.137±0.0043	-0.0010±0.0052
	PTM	-0.0029±0.0048	0.0355±0.0016	-0.0640±0.0037	0.1033±0.0081	0.1527±0.0052	0.0004±0.0037
Doc labels →	DMR	0.0091±0.0046	0.0396±0.0044	-0.0457±0.0024	0.1296±0.0085	0.1472±0.1507	0.0276±0.0101
Word features	LF-LDA	0.0130±0.0052	0.0397±0.0026	-0.0523±0.0023	0.1230±0.0153	0.1456±0.0087	0.0272±0.0042
	WF-LDA	0.0091±0.0046	0.0390±0.0051	-0.0457±0.0024	0.1296±0.0085	0.1507±0.0055	0.0276±0.0101
	GPU-DMM	-0.0934±0.0106	-0.0970±0.0034	-0.0769±0.0012	0.0836±0.0105	0.0968±0.0076	-0.0613±0.0020
Doc labels & word features →	MetaLDA	**0.0311**±0.0038	**0.0451**±0.0034	**-0.0326**±0.0019	**0.1511**±0.0093	**0.1584**±0.0072	**0.0590**±0.0065

Table V: Running time (seconds per iteration) on 80% documents of each dataset.

	Dataset	Reuters				WS				NYT	
	#Topics	50	100	150	200	50	100	150	200	200	500
No meta info	LDA	0.0899	0.1023	0.1172	0.1156	0.0219	0.0283	0.0301	0.0351	0.7509	1.1400
	PTM	4.9232	5.8885	7.2226	7.7670	1.1840	1.6375	1.8288	2.0030	-	-
Doc labels	DMR	0.6112	0.9237	1.2638	1.6066	0.4603	0.8549	1.2521	1.7173	13.7546	31.9571
	MetaLDA-dl-0.01	0.1187	0.1387	0.1646	0.1868	0.0396	0.0587	0.0769	0.112 1	2.4679	4.9928
Word features	LF-LDA	2.6895	5.3043	8.3429	11.4419	2.4920	6.0266	9.1245	11.5983	95.5295	328.0862
	WF-LDA	1.0495	1.6025	3.0304	4.8783	1.8162	3.7802	6.1863	8.6599	14.0538	31.4438
	GPU-DMM	0.4193	0.7190	1.0421	1.3229	0.1206	0.1855	0.2487	0.3118	-	-
	MetaLDA-0.1-wf	0.2427	0.4274	0.6566	0.9683	0.1083	0.1811	0.2644	0.3579	4.6205	12.4177
Doc labels & word features →	MetaLDA	0.2833	0.5447	0.7222	1.0615	0.1232	0.2040	0.3282	0.4167	6.4644	16.9735

when the word meta information is used, the SparseLDA framework no longer works in MetaLDA as the β parameters for each topic and each token are different. (2) By adapting the DistributedLDA framework [34], our MetaLDA implementation runs in parallel with multiple threads, which makes MetaLDA able to handle larger document collections. The parallel implementation was used on the NYT dataset.

The per-iteration running time of all the models is shown in Table V. Note that: (1) On the Reuters and WS datasets, all the models ran with a single thread on a desktop PC with a 3.40GHz CPU and 16GB RAM. (2) Due to the size of NYT, we report the running time for the models that are able to run in parallel. All the parallelised models ran with 10 threads on a cluster with a 14-core 2.6GHz CPU and 128GB RAM. (3) All the models were implemented in JAVA. (4) As the models with meta information add extra complexity to LDA, the per-iteration running time of LDA can be treated as the lower bound.

At the document level, both MetaLDA-df-0.01 and DMR use priors to incorporate the document meta information and both of them were implemented in the SparseLDA framework. However, our variant is about 6 to 8 times faster than DMR on the Reuters dataset and more than 10 times faster on the WS dataset. Moreover, it can be seen that the larger the number of topics, the faster our variant is over DMR. At the word level, similar patterns can be observed: our MetaLDA-0.1-wf ran significantly faster than WF-LDA and LF-LDA especially when more topics are used (20-30 times faster on WS). It is not surprising that GPU-DMM has comparable running speed with our variant, because only one topic is allowed for each document in GPU-DMM. With both document and word meta information, MetaLDA still ran several times faster than DMR, LF-LDA, and WF-LDA. On NYT with the parallel settings, MetaLDA maintains its efficiency advantage as well.

VI. Conclusion

In this paper, we have presented a topic modelling framework named MetaLDA that can efficiently incorporate document and word meta information. This gains a significant improvement over others in terms of perplexity and topic quality. With two data augmentation techniques, MetaLDA enjoys full local conjugacy, allowing efficient Gibbs sampling, demonstrated by superiority in the per-iteration running time. Furthermore, without losing generality, MetaLDA can work with both regular texts and short texts. The improvement of MetaLDA over other models that also use meta information is more remarkable, particularly when the word-occurrence information is insufficient. As MetaLDA takes a particular approach for incorporating meta information on topic models, it is possible to apply the same approach to other Bayesian probabilistic models, where Dirichlet priors are used. Moreover, it would be interesting to extend our method to use real-valued meta information directly, which is the subject of future work.

Acknowledgement

Lan Du was partially supported by Chinese NSFC project under grant number 61402312. Gang Liu was partially supported by Chinese PostDoc Fund under grant number LBH-Q15031.

References

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *JMLR*, pp. 993–1022, 2003.

[2] G. A. Miller, "WordNet: a lexical database for English," *Communications of the ACM*, pp. 39–41, 1995.

[3] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *EMNLP*, 2014, pp. 1532–1543.

[4] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionally," in *NIPS*, 2013, pp. 3111–3119.

[5] R. Das, M. Zaheer, and C. Dyer, "Gaussian LDA for topic models with word embeddings," in *ACL*, 2015, pp. 795–804.

[6] D. Q. Nguyen, R. Billingsley, L. Du, and M. Johnson, "Improving topic models with latent feature word representations," *TACL*, pp. 299–313, 2015.

[7] G. Xun, V. Gopalakrishnan, F. Ma, Y. Li, J. Gao, and A. Zhang, "Topic discovery for short texts using word embeddings," in *ICDM*, 2016, pp. 1299–1304.

[8] C. Li, H. Wang, Z. Zhang, A. Sun, and Z. Ma, "Topic modeling for short texts with auxiliary word embeddings," in *SIGIR*, 2016, pp. 165–174.

[9] D. Mimno and A. McCallum, "Topic models conditioned on arbitrary features with Dirichlet-multinomial regression," in *UAI*, 2008, pp. 411–418.

[10] D. Ramage, C. D. Manning, and S. Dumais, "Partially labeled topic models for interpretable text mining," in *SIGKDD*, 2011, pp. 457–465.

[11] J. D. Mcauliffe and D. M. Blei, "Supervised topic models," in *NIPS*, 2008, pp. 121–128.

[12] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora," in *EMNLP*, 2009, pp. 248–256.

[13] D. Kim and A. Oh, "Hierarchical Dirichlet scaling process," *Machine Learning*, pp. 387–418, 2017.

[14] C. Hu, P. Rai, and L. Carin, "Non-negative matrix factorization for discrete data with hierarchical side-information," in *AISTATS*, 2016, pp. 1124–1132.

[15] D. Andrzejewski, X. Zhu, and M. Craven, "Incorporating domain knowledge into topic modeling via Dirichlet forest priors," in *ICML*, 2009, pp. 25–32.

[16] P. Xie, D. Yang, and E. Xing, "Incorporating word correlation knowledge into topic modeling," in *NAACL*, 2015, pp. 725–734.

[17] J. Petterson, W. Buntine, S. M. Narayanamurthy, T. S. Caetano, and A. J. Smola, "Word features for Latent Dirichlet Allocation," in *NIPS*, 2010, pp. 1921–1929.

[18] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Workshop on social media analytics*, 2010, pp. 80–88.

[19] Y. Zuo, J. Wu, H. Zhang, H. Lin, F. Wang, K. Xu, and H. Xiong, "Topic modeling of short texts: A pseudo-document view," in *SIGKDD*, 2016, pp. 2105–2114.

[20] J. Yin and J. Wang, "A Dirichlet multinomial mixture model-based approach for short text clustering," in *SIGKDD*, 2014, pp. 233–242.

[21] R. Mehrotra, S. Sanner, W. Buntine, and L. Xie, "Improving LDA topic models for microblogs via tweet pooling and automatic labeling," in *SIGIR*, 2013, pp. 889–892.

[22] D. Andrzejewski, X. Zhu, M. Craven, and B. Recht, "A framework for incorporating general domain knowledge into Latent Dirichlet Allocation using first-order logic," in *IJCAI*, 2011, pp. 1171–1177.

[23] Y. Yang, D. Downey, and J. Boyd-Graber, "Efficient methods for incorporating knowledge into topic models," in *EMNLP*, 2015, pp. 308–317.

[24] H. M. Wallach, D. M. Mimno, and A. McCallum, "Rethinking LDA: Why priors matter," in *NIPS*, 2009, pp. 1973–1981.

[25] C. Chen, L. Du, and W. Buntine, "Sampling table configurations for the hierarchical Poisson-Dirichlet process," in *ECML*, 2011, pp. 296–311.

[26] Y. Teh, M. Jordan, M. Beal, and D. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, pp. 1566–1581, 2012.

[27] M. Zhou and L. Carin, "Negative binomial process count and mixture modeling," *TPAMI*, pp. 307–320, 2015.

[28] H. Zhao, L. Du, and W. Buntine, "Leveraging node attributes for incomplete relational data," in *ICML*, 2017, pp. 4072–4081.

[29] W. Buntine and M. Hutter, "A Bayesian view of the Poisson-Dirichlet process," *arXiv preprint arXiv:1007.0296v2 [math.ST]*, 2012.

[30] J. Guo, W. Che, H. Wang, and T. Liu, "Revisiting embedding features for simple semi-supervised learning," in *EMNLP*, 2014, pp. 110–120.

[31] L. Yao, D. Mimno, and A. McCallum, "Efficient methods for topic model inference on streaming document collections," in *SIGKDD*, 2009, pp. 937–946.

[32] N. Aletras and M. Stevenson, "Evaluating topic coherence using distributional semantics," in *International Conference on Computational Semantics*, 2013, pp. 13–22.

[33] J. H. Lau, D. Newman, and T. Baldwin, "Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality," in *EACL*, 2014, pp. 530–539.

[34] D. Newman, A. Asuncion, P. Smyth, and M. Welling, "Distributed algorithms for topic models," *JMLR*, pp. 1801–1828, 2009.

Knowl Inf Syst
https://doi.org/10.1007/s10115-018-1213-y

REGULAR PAPER

Leveraging external information in topic modelling

He Zhao[1] · Lan Du[1] · Wray Buntine[1] · Gang Liu[2]

Received: 20 December 2017 / Revised: 4 April 2018 / Accepted: 6 May 2018

Abstract Besides the text content, documents usually come with rich sets of meta-information, such as categories of documents and semantic/syntactic features of words, like those encoded in word embeddings. Incorporating such meta-information directly into the generative process of topic models can improve modelling accuracy and topic quality, especially in the case where the word-occurrence information in the training data is insufficient. In this article, we present a topic model called MetaLDA, which is able to leverage either document or word meta-information, or both of them jointly, in the generative process. With two data augmentation techniques, we can derive an efficient Gibbs sampling algorithm, which benefits from the fully local conjugacy of the model. Moreover, the algorithm is favoured by the sparsity of the meta-information. Extensive experiments on several real-world datasets demonstrate that our model achieves superior performance in terms of both perplexity and topic quality, particularly in handling sparse texts. In addition, our model runs significantly faster than other models using meta-information.

Keywords Latent Dirichlet allocation · Side information · Data augmentation · Gibbs sampling

Lan Du
lan.du@monash.edu

He Zhao
he.zhao@monash.edu

Wray Buntine
wray.buntine@monash.edu

Gang Liu
liugang@hrbeu.edu.cn

[1] Faculty of Information Technology, Monash University, Melbourne, VIC, Australia

[2] College of Computer Science and Technology, Harbin Engineering University, Harbin, China

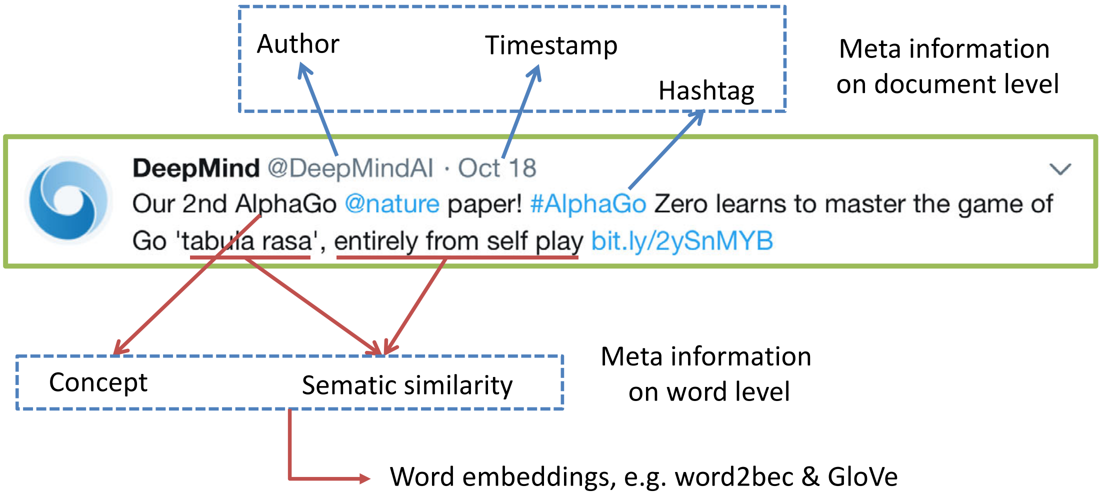
Published online: 12 May 2018

1 Introduction

With the rapid growth of the internet, huge amounts of text data are generated in social networks, online shopping and news websites, etc. These data are generally short but may contain rich and complex kinds of information that can be difficult to find in traditional information sources [44], therefore create demand for both effective and efficient machine learning techniques. Probabilistic topic models such as Latent Dirichlet Allocation (LDA) [4] are among the popular approaches for this task. In topic modelling, a document is assumed to be generated from a mixture of topics, where each topic is a probability distribution over a vocabulary. However, most existing topic models discover topics purely based on the word-occurrences, ignoring the *meta-information* (a.k.a., *side information*) associated with the content, which often results in degraded performance. We argue that meta-information associated with diverse texts can play the role of background knowledge in human text comprehension. When we humans read text, it is natural for us to leverage metadata, such as categories, authors, timestamps, words' semantic/syntactic information, to improve our understanding of the text. Therefore, it is reasonable to expect topic models can also benefit from the meta-information and yield improved modelling accuracy and topic quality.

In practice, various kinds of meta-information are associated to tweets, product reviews, blogs, etc. They are often available at both the document level and the word level. At the document level, labels of documents can be used to guide topic learning so that more meaningful topics can be discovered. It is likely that documents with common labels should discuss similar topics, which can be modelled by similar distributions over topics. In the case of tweets, as shown in Fig. 1, they can have an author, hashtag, timestamp, etc. Previous work on tweet pooling [12, 19] has shown that aggregating tweets according to their authors or hashtags can significantly improve topic modelling. Furthermore, if we use authors as labels for scientific papers, the research topics of the papers published by the same researcher can be closely related, and authors having similar research topics are more likely to collaborate [34].

At the word level, different semantic/syntactic features are also accessible. For example, there are features regarding word relationships, such as synonyms obtained from WordNet [22], word co-occurrence patterns obtained from a large corpus, and linked concepts from knowledge graphs. It is preferable that words having similar meaning but different morphological forms, like "dog" and "puppy", are likely to be assigned to the same topic,



Our 2nd AlphaGo @nature paper! #AlphaGo Zero learns to master the game of Go 'tabula rasa', entirely from self play bit.ly/2ySnMYB

Fig. 1 Meta-information associated with a tweet

even if they barely co-occur in the modelled corpus. Recently, word embeddings generated by GloVe [27] and word2vec [20,21] have attracted a lot of attention in natural language processing and related fields. It has been shown that the word embeddings can capture both the semantic and syntactic features of words so that similar words are close to each other in the embedding space. It is reasonable to expect that these word embeddings will improve topic modelling [8,26]. Figure 1 also shows some word-level meta-information associated with the tweet.

It is known that most conventional topic models can suffer from a large performance degradation on short texts (e.g., tweets and news headlines) due to insufficient word co-occurrence information. In such cases, meta-information of documents and words can play the role of auxiliary information in analysing short texts, which can compensate for the lost information in word co-occurrences. At the document level, we can leverage the hashtags, users, locations, and timestamps of tweets so that the data sparsity problem can be alleviated. At the word level, word semantic similarity and embeddings obtained or trained on large external corpus (e.g., Google News or Wikipedia) can also be built into the generative process of topic models [17,26,36].

Recently, significant research effort has been devoted to handle short texts in topic modelling. Models along this line often take classical topic models, like LDA, as a building block, and manipulate the graphical structure to incorporate meta-information into the generative process [23,26,30]. However, what we found is that those models make use of either the document level or the word level meta-information, rather than both. The limitation is often caused by their complicated model structures, which lose conjugacy favoured by sampling methods, and further result in inefficient inference algorithms.

In this article, we propose MetaLDA,[1] a new topic model that can effectively and efficiently make use of arbitrary document and word meta-information encoded in binary form. Specifically, the labels of a document in MetaLDA are incorporated in the prior of the per-document topic distributions. If two documents have similar labels, their topic distributions should be generated with similar Dirichlet priors. Analogously, at the word level, the features of a word are incorporated in the prior of the per-topic word distributions, which encourages words with similar features to have similar proportions across topics. Therefore, both document and word meta-information, if and when they are available, can be flexibly and simultaneously incorporated in the generative process. MetaLDA has the following key properties:

1. MetaLDA jointly incorporates various kinds of document and word meta-information for both regular and short texts, yielding better modelling accuracy and topic quality.
2. With data augmentation techniques, the inference of MetaLDA can be done by an efficient and closed-form Gibbs sampling algorithm that benefits from the full local conjugacy of the model.
3. The simple structure of incorporating meta-information and the efficient inference algorithm give MetaLDA advantage in terms of running speed over other models with meta-information.
4. MetaLDA has an improved interpretability. For example, the inclusion of the document labels directly in the generative process gives the ability of both explaining each label with topics and assigning labels to each topic.

We conduct extensive experiments with several real datasets including regular and short texts in various domains. The experimental results demonstrate that MetaLDA outperforms

[1] Code at https://github.com/ethanhezhao/MetaLDA/.

all the competitors we considered in terms of perplexity, topic coherence and running time. The rest of the article, which extends our earlier contribution [42], is organised as follows. We first briefly discuss the related work in Sect. 2. Then, we elaborate on MetaLDA and derive its sampling algorithm in Sects. 3 and 4, respectively. The experimental results derived on several real-world datasets are reported in Sect. 5. We conclude the article in Sect. 6.

2 Related work

In this section, we review three lines of related work: models with document meta-information, models with word meta-information, and models for short texts.

At the document level, Supervised LDA (sLDA) [18] models document labels by learning a generalised linear model with an appropriate link function and exponential family dispersion function. But the restriction for sLDA is that one document can only have one label. Labelled LDA (LLDA) [29] assumes that each label has a corresponding topic and a document is generated by a mixture of the topics. Although multiple labels are allowed in LLDA, it requires that the number of topics must equal to the number of labels, i.e., exactly one topic per label. As an extension to LLDA, Partially Labelled LDA (PLLDA) [30] relaxes this requirement by assigning multiple topics to a label. The Dirichlet Multinomial Regression (DMR) model [23] incorporates document labels on the prior of the topic distributions like our MetaLDA but with the logistic-normal transformation. As full conjugacy does not exist in DMR, a part of the inference has to be done by numerical optimisation, which is slow for large sets of labels and topics. Similarly, in the Hierarchical Dirichlet Scaling Process (HDSP) [14], conjugacy is broken as well since the topic distributions have to be renormalised. A Poisson factorisation model with hierarchical document labels is introduced in [13], but the technique cannot be applied to regular topic models as the topic proportion vectors are also unnormalised.

There has been growing interest in incorporating word features in topic models. For example, DF-LDA [2] incorporates word must-links and cannot-links using a Dirichlet forest prior in LDA; MRF-LDA [35] encodes word semantic similarity in LDA with a Markov random field; WF-LDA [28] extends LDA to model word features with the logistic-normal transform; LF-LDA [26] integrates word embeddings into LDA by replacing the topic-word Dirichlet multinomial component with a mixture of a Dirichlet multinomial component and a word embedding component; Instead of generating word types (tokens), Gaussian LDA (GLDA) [8] directly generates word embeddings with the Gaussian distribution. Despite the exciting applications of the above models, their inference is usually less efficient due to the non-conjugacy and/or complicated model structures.

Analysis of short text with topic models has been an active area with the development of social networks. Generally, there are two ways to deal with the sparsity problem in short texts, either using the intrinsic properties of short texts or leveraging meta-information. For the first way, one popular approach is to aggregate short texts into pseudo-documents, for example, [12] introduces a model that aggregates tweets containing the same word; Recently, PTM [46] aggregates short texts into latent pseudo-documents. Another approach is to assume one topic per short document, known as mixture of unigrams or Dirichlet Multinomial Mixture (DMM) such as [36,39]. For the second way, document meta-information can be used to aggregate short texts, for example, [12] aggregates tweets by the corresponding authors and [19] shows that aggregating tweets by their hashtags yields superior performance over other aggregation methods. Closely related work to ours are models that use word features for short texts. For example, [36] introduces an extension of GLDA on short texts which samples an indicator variable that chooses to generate either the type of a word or the embedding of a word and GPU-DMM [17] extends DMM with word semantic similarity obtained from embeddings for short texts. Although with improved performance, there still exist challenges for existing models:

- for aggregation-based models, it is usually hard to choose which meta-information to use for aggregation;
- the "single topic" assumption makes DMM models lose the flexibility to capture different topic ingredients of a document;
- the incorporation of meta-information in the existing models is usually less efficient.

To our knowledge, the attempts that jointly leverage document and word meta-information are relatively rare. For example, meta-information can be incorporated by first-order logic in Logit-LDA [3] and score functions in SC-LDA [37]. However, the first-order logic and score functions need to be defined for different kinds of meta-information and the definition can be infeasible for incorporating both document and word meta-information simultaneously.

3 The MetaLDA model

Given a corpus, LDA uses the same Dirichlet prior for all the per-document topic distributions and the same prior for all the per-topic word distributions [33]. While in MetaLDA, each document has a specific Dirichlet prior on its topic distribution, which is computed from the meta-information of the document, and the parameters of the prior are estimated during training. Similarly, each topic has a specific Dirichlet prior computed from the word meta-information. In this section we elaborate on our MetaLDA, in particular on how the meta-information is incorporated. Hereafter, we will use labels as document meta-information, unless otherwise stated. Table 1 summarises the notations used in this section.

The basic formulation mirrors that of standard LDA. Given a collection of D documents $\mathcal{D}$, MetaLDA generates document $d \in \{1, \ldots, D\}$ with a mixture of K topics and each topic $k \in \{1, \ldots, K\}$ is a distribution over the vocabulary with V tokens, denoted by $\boldsymbol{\phi}_k \in \mathbb{R}_+^V$. For document d with N_d words, to generate the ith ($i \in \{1, \ldots, N_d\}$) word $w_{d,i}$, we first sample a topic $z_{d,i} \in \{1, \ldots, K\}$ from the document's topic distribution $\boldsymbol{\theta_d} \in \mathbb{R}_+^K$, and then sample $w_{d,i}$ from $\boldsymbol{\phi}_{z_{d,i}}$. Now this is extended with meta-information. Assume the labels of document d are encoded in a binary vector $\boldsymbol{f_d} \in \{0, 1\}^{L_{\text{doc}}}$ where L_{doc} is the total number of unique labels. $f_{d,l} = 1$ indicates label l is active in document d and vice versa. MetaLDA allows each document to have multiple labels. Similarly, the L_{word} features of token v are stored in a binary vector $\boldsymbol{g}_v \in \{0, 1\}^{L_{\text{word}}}$. Therefore, the document and word meta-information associated with $\mathcal{D}$ are stored in the matrix $\mathbf{F} \in \{0, 1\}^{D \times L_{\text{doc}}}$ and $\mathbf{G} \in \{0, 1\}^{V \times L_{\text{word}}}$, respectively. Although MetaLDA incorporates binary features, categorical features and real-valued features can be converted into binary values with proper transformations such as discretisation and binarisation [10].

Figure 2 shows the graphical model of MetaLDA and the generative process is as follows:

1. For each topic k:

   (a) For each doc-label l: Draw $\lambda_{l,k} \sim \text{Ga}(\mu_0, \mu_0)$
   (b) For each word-feature l': Draw $\delta_{l',k} \sim \text{Ga}(\nu_0, \nu_0)$
   (c) For each token v: Compute $\beta_{k,v} = \prod_{l'=1}^{L_{\text{word}}} \delta_{l',k}^{g_{v,l'}}$
   (d) Draw $\boldsymbol{\phi}_k \sim \text{Dir}_V(\boldsymbol{\beta}_k)$

Table 1 List of notations

Notation	Description
D	Number of documents
V	Size of vocabulary
K	Number of topics
N_d	Number of words in document d
L_{doc}	Dimension of document labels
L_{word}	Dimension of word features
$\boldsymbol{f_d}$	Binary label vector of document d
$\boldsymbol{g_v}$	Binary feature vector of word v
$w_{d,i}$	ith word in document d
$z_{d,i}$	Topic of the ith word in document d
$\boldsymbol{\theta_d}$	Normalised topic weights (topic distribution) of document d
$\boldsymbol{\phi_k}$	Normalised word weights (word distribution) of topic k
$\boldsymbol{\alpha_d}$	Dirichlet parameter of the topic distribution of document d
$\boldsymbol{\beta_k}$	Dirichlet parameter of the word distribution of document k
$\lambda_{l,k}$	Weight between document label l and topic k
$\delta_{l',k}$	Weight between word feature l' and topic k
μ_0	Hyper-parameter of $\lambda_{l,k}$
ν_0	Hyper-parameter of $\delta_{l',k}$

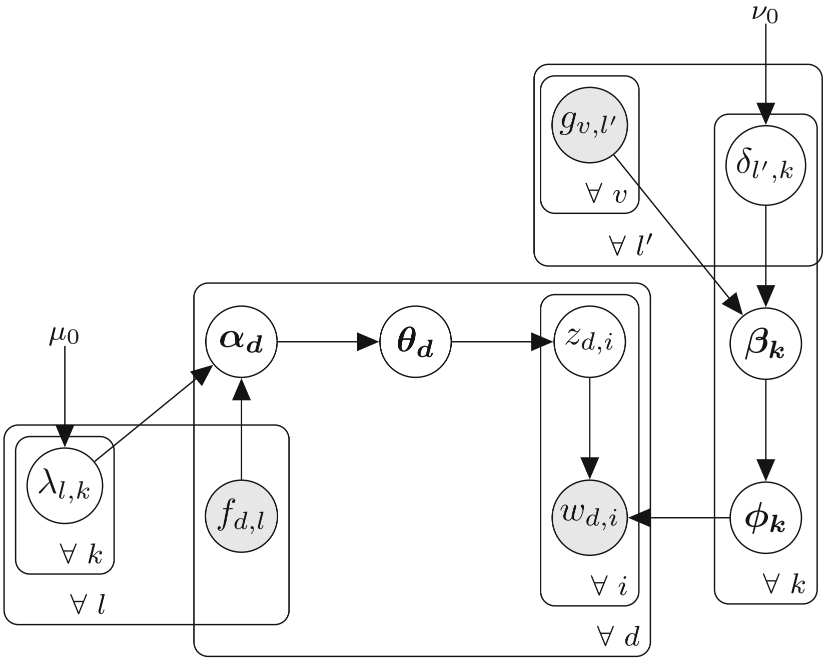


Fig. 2 The graphical model of MetaLDA

2. For each document d:
    (a) For each topic k: Compute $\alpha_{d,k} = \prod_{l=1}^{L_{\text{doc}}} \lambda_{l,k}^{f_{d,l}}$
    (b) Draw $\boldsymbol{\theta}_d \sim \text{Dir}_K(\boldsymbol{\alpha}_d)$
    (c) For each word in document d:
        (i) Draw topic $z_{d,i} \sim \text{Cat}_K(\boldsymbol{\theta}_d)$
        (ii) Draw word $w_{d,i} \sim \text{Cat}_V(\boldsymbol{\phi}_{z_{d,i}})$

where Ga(·, ·), Dir(·), Cat(·) are the gamma distribution with shape and rate parameters, the Dirichlet distribution, and the categorical distribution, respectively. K, μ_0, and ν_0 are the hyper-parameters.

To incorporate document labels, MetaLDA learns a specific Dirichlet prior over the topics for each document by using the label information. Specifically, the information of document d's labels is incorporated in $\boldsymbol{\alpha}_d$, the parameter of Dirichlet prior on $\boldsymbol{\theta}_d$. As shown in Step 2a, $\alpha_{d,k}$ is computed as a log linear combination of the labels $f_{d,l}$. Since $f_{d,l}$ is binary, $\alpha_{d,k}$ is indeed the multiplication of $\lambda_{l,k}$ over all the active labels of document d, i.e., $\{l \mid f_{d,l} = 1\}$. Drawn from the gamma distribution with mean 1, $\lambda_{l,k}$ controls the impact of label l on topic k. If label l has no or less impact on topic k, $\lambda_{l,k}$ is expected to be 1 or close to 1, and then $\lambda_{l,k}$ will have no or little influence on $\alpha_{d,k}$ and vice versa. The hyper-parameter μ_0 controls the variation of $\lambda_{l,k}$. The incorporation of word features is analogous but in the parameter of the Dirichlet prior on the per-topic word distributions as shown in Step 1c.

The intuition of our way of incorporating meta-information is as follows. At the document level, if two documents have more labels in common, their Dirichlet parameter $\boldsymbol{\alpha}_d$ will be more similar, resulting in more similar topic distributions $\boldsymbol{\theta}_d$; At the word level, if two words have similar features, their $\beta_{k,v}$ in topic k will be similar and then we can expect that their $\phi_{k,v}$ could be more or less the same. Finally, the two words will have similar probabilities of showing up in topic k. In other words, if a topic "prefers" a certain word, we expect that it will also prefer other words with similar features to that word. Moreover, at both the document and the word level, different labels/features may have different impact on the topics (λ/δ), which can be automatically learnt in MetaLDA from the data.

4 Inference

Unlike most existing methods, our way of incorporating the meta-information facilitates the derivation of an efficient Gibbs sampling algorithm. With two data augmentation techniques (i.e., the introduction of auxiliary variables), MetaLDA admits the local conjugacy that further gives us a close-form Gibbs sampling algorithm. Note that MetaLDA incorporates the meta-information on the Dirichlet priors, so we can still use LDA's collapsed Gibbs sampling algorithm for the topic assignment $z_{d,i}$. Thus, there is no need to use a hybrid learning algorithm (i.e., optimisation + sampling), such as those in [23,26]. Moreover, as shown in Step 2a and 1c, we only need to consider nonzero entries of $\mathbf{F}$ and $\mathbf{G}$ in computing the full conditionals, which further reduces the inference complexity, particularly when the feature space is sparse. This is often the case in real-world scenarios. In the rest of this section, we will focus on the derivation of the full conditionals for sampling the two Gamma random variables, $\boldsymbol{\lambda}$ and $\boldsymbol{\delta}$, used to modelling the influence of document labels and word features on topics. Table 2 shows the statistics that we need while running the inference.

Table 2 Summary of statistics

Notation	Description
$m_{d,k}$	Number of words in document d assigned to topic k
$n_{k,v}$	Number of word v assigned to topic k
q_d	Beta distributed axillary variable for document d
$t_{d,k}$	Axillary table counts drawn from CRP for document d and topic k
$\hat{q}_k$	Beta distributed axillary variable for topic k
$t'_{d,k}$	Axillary table counts drawn from CRP for document k and word v

Given $\boldsymbol{\phi}_{1:K}$ and $\boldsymbol{\theta}_{1:D}$, the complete model likelihood (i.e., joint distribution) of MetaLDA is exactly the same as LDA's likelihood, which is as follows:

$$\Pr(\boldsymbol{w}_{1:D}, \boldsymbol{z}_{1:D} | \boldsymbol{\theta}_{1:D}, \boldsymbol{\phi}_{1:K}) = \prod_{d=1}^{D} \prod_{i=1}^{N_d} \theta_{d,z_{d,i}} \phi_{z_{d,i},v} = \prod_{d=1}^{D} \prod_{k=1}^{K} \theta_{d,k}^{m_{d,k}} \prod_{k=1}^{K} \prod_{v=1}^{V} \phi_{k,v}^{n_{k,v}} \tag{1}$$

where $n_{k,v} = \sum_{d}^{D} \sum_{i=1}^{N_d} \mathbf{1}_{(w_{d,i}=v, z_{d,i}=k)}$ counts the number of words v assigned to topic k, $m_{d,k} = \sum_{i=1}^{N_d} \mathbf{1}_{(z_{d,i}=k)}$ counts the number of words in document d assigned to topic k, and $\mathbf{1}_{(\cdot)}$ is the indicator function. In the standard LDA model, we can marginalise out $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ using the Dirichlet multinomial conjugacy, and then yield

$$\begin{aligned}
&\Pr(\boldsymbol{z}_{1:D}, \boldsymbol{w}_{1:D}; \boldsymbol{\alpha}_{1:D}, \boldsymbol{\beta}_{1:K}) \\
&= \int_{\boldsymbol{\theta}} \prod_{d=1}^{D} \frac{\Gamma\left(\sum_{k=1}^{K} \alpha_{d,k}\right)}{\prod_{k=1}^{K} \Gamma(\alpha_{d,k})} \prod_{k=1}^{K} \theta_{d,k}^{m_{d,k}+\alpha_{d,k}-1} \int_{\boldsymbol{\phi}} \prod_{k=1}^{K} \frac{\Gamma\left(\sum_{v=1}^{V} \beta_{k,v}\right)}{\prod_{v=1}^{V} \Gamma(\beta_{k,v})} \prod_{v=1}^{V} \phi_{k,v}^{n_{k,v}+\beta_{k,v}-1} . \\
&= \prod_{d=1}^{D} \frac{\mathrm{Beta}_K(\boldsymbol{\alpha}_d + \boldsymbol{m}_d)}{\mathrm{Beta}_K(\boldsymbol{\alpha}_d)} \prod_{k=1}^{K} \frac{\mathrm{Beta}_V(\boldsymbol{\beta}_k + \boldsymbol{n}_k)}{\mathrm{Beta}_V(\boldsymbol{\beta}_k)}
\end{aligned} \tag{2}$$

where $\Gamma(\cdot)$ is the Gamma function, $\mathrm{Beta}_N(\cdot)$ is a N-dimensional beta function as

$$\mathrm{Beta}_N(\boldsymbol{x}) = \frac{\prod_n \Gamma(x_n)}{\Gamma\left(\sum_n x_n\right)}$$

and here we assume that the Dirichlet priors are document and topic specific. Given $\boldsymbol{\beta}_k$ and $\boldsymbol{\alpha}_d$, it is straightforward to compute the full conditional for sampling topic assignment $z_{d,i}$, i.e.,

$$\begin{aligned}
\Pr(z_{d,i} = k \mid \boldsymbol{z}_{1:D}^{-z_{d,i}}, \boldsymbol{w}_{1:D}, \boldsymbol{\alpha}_{\mathbf{1:D}}, \boldsymbol{\beta}_{\mathbf{1:K}}) &= \frac{\Pr(z_{d,i} = k, \boldsymbol{z}_{1:D}^{-z_{d,i}}, \boldsymbol{w}_{1:D}, \boldsymbol{\alpha}_{1:D}, \boldsymbol{\beta}_{1:K})}{\Pr(\boldsymbol{z}_{1:D}^{-z_{d,i}}, \boldsymbol{w}_{1:D}, \boldsymbol{\alpha}_{1:D}, \boldsymbol{\beta}_{1:K})} \\
&\propto (\alpha_{d,k} + m_{d,k}) \frac{\beta_{k,v} + n_{k,v}}{\beta_{k,\cdot} + n_{k,\cdot}} .
\end{aligned} \tag{3}$$

In MetaLDA, we have replaced $\boldsymbol{\alpha}_d$ and $\boldsymbol{\beta}_k$ with a log linear model in order to build informative priors from various side information associated with both documents and words. They are deterministically computed from a set of Gamma random variables, as shown in Step 2a and 1c in the generative process. Equation (3) can still be used in MetaLDA to sample the topic assignments. However, the major challenge is to sample the Gamma random variables, $\boldsymbol{\lambda}$ and $\boldsymbol{\delta}$ without significantly complicating the inference procedure.

4.1 Sampling Gamma random variable $\lambda_{l,k}$

$\lambda_{l,k}$ is involved in computing the Dirichlet prior over $\boldsymbol{\theta}_{1:D}$ via the parameter $\boldsymbol{\alpha}_{1:D}$. To sample $\lambda_{l,k}$, we expand the first Beta ratio in Eq. (2) with Gamma functions as follows:

$$\prod_{d=1}^{D} \frac{\mathrm{Beta}_K(\boldsymbol{\alpha}_d + \boldsymbol{m}_d)}{\mathrm{Beta}_K(\boldsymbol{\alpha}_d)} = \prod_{d=1}^{D} \underbrace{\frac{\Gamma(\alpha_{d,\cdot})}{\Gamma(\alpha_{d,\cdot} + m_{d,\cdot})}}_{\text{Gamma ratio 1}} \prod_{k=1}^{K} \underbrace{\frac{\Gamma(\alpha_{d,k} + m_{d,k})}{\Gamma(\alpha_{d,k})}}_{\text{Gamma ratio 2}} \tag{4}$$

where $\alpha_{d,\cdot} = \sum_{k=1}^{K} \alpha_{d,k}$, and $m_{d,\cdot} = \sum_{k=1}^{K} m_{d,k}$. It is not easy to directly work with these Gamma functions, while we replace α_k with $\prod_{l=1}^{L_{\mathrm{doc}}} \lambda_{l,k}^{f_{d,l}}$. In order to retain the sampling efficiency of the standard LDA model, we appeal to data augmentation.

Gamma ratio 1 in Eq. (4) can be seen to be the marginalisation of a set of Beta random variables, therefore can be augmented as (similar to the sampling of the Pitman–Yor concentration parameter in [9]):

$$\underbrace{\frac{\Gamma(\alpha_{d,\cdot})}{\Gamma(\alpha_{d,\cdot} + m_{d,\cdot})}}_{\text{Gamma ratio 1}} \propto \int_{q_d} q_d^{\alpha_{d,\cdot}-1} (1 - q_d)^{m_{d,\cdot}-1} \tag{5}$$

where for each document d, $q_d \sim \mathrm{Beta}(\alpha_{d,\cdot}, m_{d,\cdot})$. Given a set of $q_{1:D}$ for all the documents, Gamma ratio 1 can be approximated by the product of $q_{1:D}$, i.e., $\prod_{d=1}^{D} q_d^{\alpha_{d,\cdot}}$.

Gamma ratio 2 in Eq. (4) is the Pochhammer symbol for a rising factorial, which can be augmented with an auxiliary variable $t_{d,k}$ [7,31,40,45] as follows:

$$\underbrace{\frac{\Gamma(\alpha_{d,k} + m_{d,k})}{\Gamma(\alpha_{d,k})}}_{\text{Gamma ratio 2}} = \sum_{t_{d,k}=0}^{m_{d,k}} S_{t_{d,k}}^{m_{d,k}} \alpha_{d,k}^{t_{d,k}} \tag{6}$$

where S_t^m indicates an unsigned Stirling number of the first kind. Gamma ratio 2 is indeed a normalising constant for the probability of the number of tables in the Chinese Restaurant Process (CRP) [5], $t_{d,k}$ can be sampled by a CRP with $\alpha_{d,k}$ as the concentration and $m_{d,k}$ as the number of customers:

$$t_{d,k} = \sum_{i=1}^{m_{d,k}} \mathrm{Bern}\left(\frac{\alpha_{d,k}}{\alpha_{d,k} + i}\right) \tag{7}$$

where $\mathrm{Bern}(\cdot)$ samples a sequence of binary variables from the Bernoulli distribution. The complexity of sampling $t_{d,k}$ by Eq. (7) is $\mathcal{O}(m_{d,k})$. For large $m_{d,k}$, as the standard deviation of $t_{d,k}$ is $\mathcal{O}(\sqrt{\log m_{d,k}})$ [5], one can sample $t_{d,k}$ in a small window around the current value in complexity $\mathcal{O}(\sqrt{\log m_{d,k}})$.

By ignoring the terms unrelated to α, the augmentation of Eq. (6) can be simplified to a single term $\alpha_{d,k}^{t_{d,k}}$. With those auxiliary variables, we can simplify Eq. (4) as:

$$\prod_{d=1}^{D} q_d^{\alpha_{d,\cdot}} \prod_{k=1}^{K} \alpha_{d,k}^{t_{d,k}} = \prod_{d=1}^{D} \prod_{k=1}^{K} q_d^{\alpha_{d,k}} \alpha_{d,k}^{t_{d,k}} \tag{8}$$

Now, replacing $\alpha_{d,k}$ with $\lambda_{l,k}$ (i.e., $\alpha_{d,k} = \prod_{l=1}^{L_{\text{doc}}} \lambda_{l,k}^{f_{d,l}}$), we get:

$$\begin{aligned}
&\left(\prod_{d=1}^{D}\prod_{k=1}^{K} e^{\alpha_{d,k}\log q_d}\right)\left(\prod_{d=1}^{D}\prod_{k=1}^{K}\left(\prod_{l=1}^{L_{\text{doc}}}\lambda_{l,k}^{f_{d,l}}\right)^{t_{d,k}}\right)\\
&=\left(\prod_{d=1}^{D}\prod_{k=1}^{K} e^{-\alpha_{d,k}\log\frac{1}{q_d}}\right)\left(\prod_{l=1}^{L_{\text{doc}}}\prod_{k=1}^{K}\lambda_{l,k}^{\sum_{d=1}^{D} f_{d,l}t_{d,k}}\right)\\
&=\left(\prod_{k=1}^{K} e^{-\sum_{d=1}^{D}\alpha_{d,k}\log\frac{1}{q_d}}\right)\left(\prod_{l=1}^{L_{\text{doc}}}\prod_{k=1}^{K}\lambda_{l,k}^{\sum_{d=1}^{D} f_{d,l}t_{d,k}}\right)
\end{aligned} \tag{9}$$

Recall that all the document labels are binary and $\lambda_{l,k}$ is involved in computing $\alpha_{d,k}$ if and only if $f_{d,l} = 1$. Extracting all the terms related to $\lambda_{l,k}$ in Eq. (9), we get the posterior likelihood of $\lambda_{l,k}$:

$$e^{-\lambda_{l,k}\left(\sum_{d=1:f_{d,l}=1}^{D}\frac{\alpha_{d,k}}{\lambda_{l,k}}\log\frac{1}{q_d}\right)}\lambda_{l,k}^{\sum_{d=1}^{D} f_{d,l}t_{d,k}}$$

where $\frac{\alpha_{d,k}}{\lambda_{l,k}}$ is the value of $\alpha_{d,k}$ with $\lambda_{l,k}$ removed when $f_{d,l} = 1$. With these data augmentation techniques, the likelihood is transformed into a form that is conjugate to the gamma prior of $\lambda_{l,k}$.

$$\begin{aligned}
\Pr(\lambda_{l,k}) &\propto e^{-\lambda_{l,k}\left(\sum_{d=1:f_{d,l}=1}^{D}\frac{\alpha_{d,k}}{\lambda_{l,k}}\log\frac{1}{q_d}\right)}\lambda_{l,k}^{\sum_{d=1}^{D} f_{d,l}t_{d,k}}\lambda_{l,k}^{\mu_0-1}e^{-\lambda_{l,k}\mu_0}\\
&= e^{-\lambda_{l,k}\left(\mu_0-\sum_{d=1:f_{d,l}=1}^{D}\frac{\alpha_{d,k}}{\lambda_{l,k}}\log q_d\right)}\lambda_{l,k}^{\mu_0+\sum_{d=1}^{D} f_{d,l}t_{d,k}-1}
\end{aligned}$$

Therefore, it is straightforward to yield the following sampling strategy for $\lambda_{l,k}$:

$$\lambda_{l,k} \sim \mathrm{Ga}(\mu', \mu'') \tag{10}$$

$$\mu' = \mu_0 + \sum_{d=1:f_{d,l}=1}^{D} t_{d,k} \tag{11}$$

$$\mu'' = \mu_0 - \sum_{d=1:f_{d,l}=1}^{D} \frac{\alpha_{d,k}}{\lambda_{l,k}}\log q_d \tag{12}$$

Before $\lambda_{l,k}$ is sampled, the value of $\alpha_{d,k}$ can be computed and cached. After a new value of $\lambda_{l,k}$ is sampled, $\alpha_{d,k}$ is updated by:

$$\alpha_{d,k} \leftarrow \frac{\alpha_{d,k}\lambda'_{l,k}}{\lambda_{l,k}},\ \forall\, 1 \le d \le D : f_{d,l} = 1 \tag{13}$$

where $\lambda'_{i,k}$ is the newly sampled value of $\lambda_{i,k}$.

To sample/compute Eqs. (10)–(13), one only iterates over the documents where label l is active (i.e., $f_{d,l} = 1$). Thus, the sampling for all λ takes $\mathcal{O}(D'KL_{\text{doc}})$ where D' is the average number of documents where a label is active (i.e., the column-wise sparsity of $\mathbf{F}$). It is usually that $D' \ll D$ because if a label exists in nearly all the documents, it provides little discriminative information and can then be neglected. This demonstrates how the sparsity of document meta-information is leveraged. Moreover, sampling all the tables t takes $\mathcal{O}(\tilde{N})$ ($\tilde{N}$ is the total number of words in $\mathcal{D}$) which can be accelerated with the window sampling technique explained above.

4.2 Sampling Gamma random variable $\delta_{l',k}$

The derivation of sampling $\delta_{l',k}$ is analogous to $\lambda_{l,k}$. Here, we use the same data augmentation methods for re-parameterising the second Beta ratio in Eq. (2), i.e.,

$$\prod_{k=1}^{K} \frac{\mathrm{Beta}_V(\boldsymbol{\beta}_k + \boldsymbol{n}_k)}{\mathrm{Beta}_V(\boldsymbol{\beta}_k)} = \prod_k \frac{\Gamma(\beta_{k,\cdot})}{\Gamma(\beta_{k,\cdot} + n_{k,\cdot})} \prod_v \frac{\Gamma(\beta_{k,v} + n_{k,v})}{\Gamma(\beta_{k,v})} \tag{14}$$

as

$$\prod_{k=1}^{K} \prod_{v=1}^{V} \hat{q}_k^{\beta_{k,v}} \beta_{k,v}^{t'_{k,v}} \tag{15}$$

where $\hat{q}_k \sim \mathrm{Be}(\beta_{k,\cdot}, n_{k,\cdot})$ and $t'_{k,v} = \sum_{i=1}^{n_{k,v}} \mathrm{Bern}\left(\frac{\beta_{k,v}}{\beta_{k,v}+i}\right)$. Now, we replace $\beta_{k,v}$ with $\prod_{l'=1}^{L_{\text{word}}} \delta_{l',k}^{g_{v,l'}}$,

$$\begin{aligned} &\left(\prod_{k=1}^{K} \prod_{v=1}^{V} e^{-\delta_{l',k} \frac{\beta_{k,v}}{\delta_{l',k}} \log \frac{1}{\hat{q}_k}}\right) \left(\prod_{k=1}^{K} \prod_{v=1}^{V} \left(\prod_{l'=1}^{L_{\text{word}}} \delta_{l',k}^{g_{v,l'}}\right)^{t'_{k,v}}\right) \\ &= \prod_{k=1}^{K} e^{-\delta_{l',k} \left(\sum_{v=1}^{V} \frac{\beta_{k,v}}{\delta_{l',k}}\right) \log \frac{1}{\hat{q}_k}} \left(\prod_{k=1}^{K} \prod_{l'=1}^{L_{\text{word}}} \delta_{l',k}^{\sum_{v=1}^{V} g_{v,l'} t'_{k,v}}\right) \end{aligned}$$

and then extract all the terms related to $\delta_{l',k}$ in Eq. (15), and add the Gamma prior, we derive the posterior of $\delta_{l',k}$:

$$\Pr(\delta_{l',k}) \propto e^{-\delta_{l',k}\left(\nu_0 - \log \hat{q}_k \sum_{v=1:g_{v,l'}=1}^{V} \frac{\beta_{k,v}}{\delta_{l',k}}\right)} \delta_{l',k}^{\nu_0 + \sum_v g_{v,l'} t'_{k,v} - 1}$$

We can then sample $\delta_{l',k}$ from a Gamma distribution parameterised with

$$\delta_{l',k} \sim \mathrm{Ga}(\nu', \nu'') \tag{16}$$

$$\nu' = \nu_0 + \sum_{v=1:g_{v,l'}=1}^{V} t'_{k,v} \tag{17}$$

$$\nu'' = \nu_0 - \log \hat{q}_k \sum_{v=1:g_{v,l'}=1}^{V} \frac{\beta_{k,v}}{\delta_{l',k}} \tag{18}$$

$\beta_{k,v}$ can be updated in a similar way to $\alpha_{d,k}$, i,e,

$$\beta_{k,v} \leftarrow \frac{\beta'_{k,v} \delta'_{l',k}}{\delta_{l',k}}, \ \forall\, 1 \le k \le K : g_{v,l'} = 1 \tag{19}$$

where $\delta'_{l',k}$ is newly sampled value of $\delta_{l',k}$. Sampling all δ takes $\mathcal{O}(V' K L_{\text{word}})$ where V' is the average number of tokens where a feature is active (i.e., the column-wise sparsity of $\mathbf{G}$ and usually $V' \ll V$) and sampling all the tables t' takes $\mathcal{O}(\tilde{N})$. Figure 3 illustrates the full sampling algorithm.

```
Require: $\mathcal{D}$, $\mathbf{F}$ (if available), $\mathbf{G}$ (if available), $K$, $\mu_0$, $\nu_0$, $MaxIteration$
Ensure: topic assignments for all words: $z_{d,i}$
Randomly initialise $z_{d,i}$, $\lambda_{l,k}$ (Step 1a), $\delta_{l',k}$ (Step 1b)
Compute $\alpha_{d,k}$ (Step 2a), $\beta_{k,v}$ (Step 1c), $m_{d,k}$, $n_{k,v}$
for $iter \leftarrow 1$ to $MaxIteration$ do
    for all document $d$ do
        for all word $w_{d,i} = v$ $(z_{d,i} = k)$ in $d$ do
            $m_{d,k} = m_{d,k} - 1$, $n_{k,v} = n_{k,v} - 1$
            Sample new topic $k'$ according to Eq. (3)
            $z_{d,i} = k'$, $m_{d,k'} = m_{d,k'} + 1$, $n_{k',v} = n_{k',v} + 1$
        end for
    end for
    for all document $d$ do
        Sample $q_d$ by $q_d \sim \mathrm{Beta}(\alpha_{d,\cdot}, m_{d,\cdot})$
        for all topic $k$ do
            Sample $t_{d,k}$ according to Eq. (7)
        end for
    end for
    for all document label $l$ and topic $k$ do
        Sample $\lambda_{l,k}$ according to Eq. (10) to Eq. (12)
        Update $\alpha_{d,k}$ according to Eq. (13)
    end for
    for all topic $k$ do
        Sample $\hat{q}_k$ by $\hat{q}_k \sim \mathrm{Beta}(\beta_{k,\cdot}, n_{k,\cdot})$
        for all word $v$ do
            Sample $t'_{k,v}$ by $t'_{k,v} = \sum_{i=1}^{n_{k,v}} \mathrm{Bern}\left(\frac{\beta_{k,v}}{\beta_{k,v}+i}\right)$
        end for
    end for
    for all word feature $l'$ and topic $k$ do
        Sample $\delta_{l',k}$ according to Eq. (16) to Eq. (18)
        Update $\beta_{k,v}$ according to Eq. (19)
    end for
end for
```

Fig. 3 Collapsed Gibbs sampling algorithm for MetaLDA

4.3 MetaLDA as a hyper-parameter sampling approach

Besides the observed labels/features associated with the datasets, a default label/feature for each document/word is introduced in MetaLDA, which is always equal to 1. The default can be interpreted as the bias term in α/β, which is supposed to capture the information unrelated to the labels/features. When working without document labels with the default, MetaLDA samples the Dirichlet parameters (i.e., Hyper-parameters of LDA) of the document-topic distributions, α, according to the statistics in the target corpus. Similarly, without word features, the Dirichlet parameters of the topic-word distributions, β, are sampled. We demonstrate this by taking the document-topic distributions as an example.

Now assume each document only has a default label that is always equal to 1, i.e., $f_{d,0} = 1$ and $f_{d,l} = 0$ for all $l > 0$. According to our construction (Step 1 and 2a), $a_{d,k} = \lambda_{0,k}$ for all the document. In other words, all the documents share the same asymmetric Dirichlet prior on the document-topic distributions ($\boldsymbol{\theta}_d$) which is constructed as follows:

$$\alpha_k \sim \mathrm{Ga}(\mu_0, \nu_0) \tag{20}$$

$$\boldsymbol{\theta}_d \sim \mathrm{Dir}_K(\boldsymbol{\alpha}) \tag{21}$$

In this case, we can sample α_k as follows:

$$\alpha_k \sim \mathrm{Ga}\left(\mu_0 + t_{\cdot,k}, \mu_0 - \sum_{d=1}^{D} \log q_d\right) \tag{22}$$

Alternatively, we can vary MetaLDA to have a **symmetric Dirichlet prior**:

$$\alpha \sim \mathrm{Ga}(\mu_0, \mu_0) \tag{23}$$

$$\boldsymbol{\theta}_d \sim \mathrm{Dir}_K(\alpha, \ldots, \alpha) \tag{24}$$

In this case, we can sample α as follows:

$$\alpha \sim \mathrm{Ga}\left(\mu_0 + t_{\cdot,\cdot}, \mu_0 - \sum_{d=1}^{D} \log q_d\right) \tag{25}$$

Discussed in [6,33], sampling the Dirichlet priors can gain significant performance improvement in topic models. In the case where document labels/word features are not used, MetaLDA offers an alternative hyper-parameter sampling approach to the methods such as fixed-point iterations [24] and Newton–Raphson [32]. These methods use MAP to optimise the hyper-parameters while ours uses MCMC sampling. We would like to point out that MetaLDA's sampling of symmetric Dirichlet prior is similar to the approach introduced in [31]. However the sampling of asymmetric prior was not considered in [31]. Compared with the built-in hyper-parameter sampling methods in Mallet[2] which are based on histograms of the statistics, our approach is more robust in the case where the statistics are not sufficient (e.g., short texts). This is further discussed with experiments in Sect. 5.4.3.

5 Experiments

In this section, we evaluate the proposed MetaLDA against several recent alternatives that also incorporate meta-information, using 6 real datasets including both regular and short texts. We will focus on the evaluation of

- the modelling accuracy of MetaLDA in terms of perplexity, a standard measure used in topic modelling. The goal is to study how the meta-information contributes to the predictive likelihood of unseen documents.
- the quality of topics learned by MetaLDA. It is interesting to see whether or not the meta-information will positively affect the topic coherence. We will report both quantitative and qualitative analyses.
- the running time of MetaLDA. The introduction of meta-information increases the modelling complexity to some extend. However, as we discussed in previous sections, MetaLDA can benefit from the local conjugacy given by the data augmentation methods, and also be parallelised using the same distributed framework [25] in Mallet. Therefore, we will empirically study the efficiency of MetaLDA.

[2] http://mallet.cs.umass.edu.

Besides, we will also study how word embeddings learnt by different techniques affect both perplexity and topic coherence.

5.1 Datasets

In the experiments, we used three regular and three short text datasets, which are as follows:

- *Reuters* is a widely used corpus extracted from the Reuters-21578 dataset where documents without any labels are removed.[3] There are 11,367 documents and 120 labels. Each document is associated with multiple labels. The vocabulary size is 8817, and the average document length is 73.
- *20NG* 20 Newsgroups is a widely used dataset consists of 18,846 news articles with 20 categories. The vocabulary size is 22,636 and the average document length is 108.
- *NYT* New York Times is extracted from the documents in the category "Top/News/Health" in the New York Times Annotated Corpus.[4] There are 52,521 documents and 545 unique labels. Each document is with multiple labels. The vocabulary contains 21,421 tokens, and there are 442 words in a document on average.
- *WS* Web Snippets, used in [17], contains 12,237 web search snippets and each snippet belongs to one of 8 categories. The vocabulary contains 10,052 tokens, and there are 15 words in one snippet on average.
- *TMN* Tag My News, used in [26], consists of 32,597 English RSS news snippets from Tag My News. With a title and a short description, each snippet belongs to one of 7 categories. There are 13,370 tokens in the vocabulary, and the average length of a snippet is 18.
- *AN* ABC News, is a collection of 12,495 short news descriptions and each one is in multiple of 194 categories. There are 4255 tokens in the vocabulary, and the average length of a description is 13.

All the datasets were tokenised by Mallet (see footnote 2) and we removed the words that exist in less than 5 documents and more than 95% of the documents.

5.2 Meta-information settings

At the document level, the labels associated with documents in each dataset were used as the meta-information. At the word level, we used a set of binarised word embeddings as word features (see footnote 3), which are obtained from real-valued word embeddings such as GloVe or word2vec. To binarise word embeddings, we first adopted the following method similar to [11]:

$$g'_{v,j} = \begin{cases} 1, & \text{if } g''_{v,j} > \text{Mean}_+(\boldsymbol{g}''_v) \\ -1, & \text{if } g''_{v,j} < \text{Mean}_-(\boldsymbol{g}''_v) \\ 0, & \text{otherwise} \end{cases} \tag{26}$$

where $\boldsymbol{g}''_v$ is the original embedding vector for word v, $g'_{v,j}$ is the binarised value for jth element of $\boldsymbol{g}''_v$, and $\text{Mean}_+(\cdot)$ and $\text{Mean}_-(\cdot)$ are the average value of all the positive elements and negative elements, respectively.

[3] MetaLDA is able to handle documents/words without labels/features. But for fair comparison with other models, we removed the documents without labels and words without features.

[4] https://catalog.ldc.upenn.edu/ldc2008t19.

The insight is that we only consider features with strong opinions (i.e., large positive or negative value) on each dimension. To transform $g' \in \{-1, 1\}$ to the final $g \in \{0, 1\}$, we use two binary bits to encode one dimension of $g'_{v,j}$: the first bit is on if $g'_{v,j} = 1$ and the second is on if $g'_{v,j} = -1$. This means that if the original embeddings are 100-dimensional, the binarised embeddings will be with 200 dimensions. In our experiments, we also tried some other word embedding binarisation methods including the one in [10]. However, the performance with those binarisation methods is not comparable with the one we proposed above. Therefore, the experimental results with different binarisation methods will not be reported.

In the perplexity and topic coherence evaluation, i.e., Sects. 5.4 and 5.5, we will use the 50-dimensional GloVe word embeddings pre-trained on Wikipedia[5] as the source of word features. We then study how different word embedding sources influence the performance of our model in Sect. 5.6. It is noteworthy that MetaLDA can also work with other word features such as semantic similarity.

5.3 Compared models and parameter settings

We evaluate the performance of the following models:

– *MetaLDA* and its variants: the proposed model and its variants. Here we use MetaLDA to indicate the model considering both document labels and word features. Several variants of MetaLDA with document labels and word features separately were also studied, which are shown in Table 3. These variants differ in the method of estimating $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. All the models listed in Table 3 were implemented on top of Mallet. The hyper-parameters μ_0 and ν_0 were set to 1.0.
– *LDA* [4]: the baseline model. The Mallet implementation of SparseLDA [38] is used.
– *LLDA*, Labelled LDA [29] and **PLLDA**, Partially Labelled LDA [30]: two models that make use of multiple document labels. The original implementation[6] is used.
– *DMR*, LDA with Dirichlet Multinomial Regression [23]: a model that can use multiple document labels. The Mallet implementation of DMR based on SparseLDA was used. Following Mallet, we set the mean of λ to 0.0 and set the variances of λ for the default label and the document labels to 100.0 and 1.0, respectively.
– *WF-LDA*, Word Feature LDA [28]: a model with word features. We implemented it on top of Mallet and used the default settings in Mallet for the optimisation.
– *LF-LDA*, Latent Feature LDA [26]: a model that incorporates word embeddings. The original implementation[7] was used. Following the original paper, we used 1500 and 500 MCMC iterations for initialisation and sampling, respectively, and set λ to 0.6, and used the original 50-dimensional GloVe word embeddings as word features.
– *GPU-DMM*, Generalized Pólya Urn DMM [17]: a model that incorporates word semantic similarity. The original implementation[8] was used. The word similarity was generated from the distances of the word embeddings. Following the original paper, we set the hyper-parameters μ and ϵ to 0.1 and 0.7, respectively, and the symmetric document Dirichlet prior to $50/K$.

[5] https://nlp.stanford.edu/projects/glove/.

[6] https://nlp.stanford.edu/software/tmt/tmt-0.4/.

[7] https://github.com/datquocnguyen/LFTM.

[8] https://github.com/NobodyWHU/GPUDMM.

Table 3 MetaLDA and its variants

	Compute α with	Compute β with
MetaLDA	Document labels	Word features
MetaLDA-dl-def	Document labels	Default feature
MetaLDA-dl-0.01	Document labels	Symmetric 0.01 (fixed)
MetaLDA-def-wf	Default label	Word features
MetaLDA-0.1-wf	Symmetric 0.1 (fixed)	Word features
MetaLDA-def-def	Default label	Default feature

Table 4 Summary of the compared models

Meta Info used	Model
None	LDA [4]
	PTM [46]
	MetaLDA-def-def
Document labels	LLDA [29]
	PLLDA [30]
	DMR [23]
	MetaLDA-dl-def
	MetaLDA-dl-0.01
Word features	WF-LDA [28]
	LF-LDA [26]
	MetaLDA-def-wf
	MetaLDA-0.1-wf
	GPU-DMM [17]
Both	MetaLDA

- *PTM*, Pseudo document based Topic Model [46]: a model for short text analysis. The original implementation[9] was used. Following the paper, we set the number of pseudo-documents to 1000 and λ to 0.1.

All the models, except where noted, the symmetric parameters of the document and the topic Dirichlet priors were set to 0.1 and 0.01, respectively, and 2000 MCMC iterations are used to train the models. We summarise the compared models in terms of their usage of meta-information in Table 4.

5.4 Perplexity evaluation

Perplexity is a measure that is widely used [33] to evaluate the modelling accuracy of topic models. The lower the score, the higher the modelling accuracy. To compute perplexity, we randomly selected some documents in a dataset as the training set and the remaining as the test set. We first trained a topic model on the training set to get the word distributions of each topic k ($\boldsymbol{\phi}_k^{\text{train}}$). Each test document d was split into two halves containing every first and every second word, respectively. We then fixed the topics and trained the models on the first half to get the topic proportions ($\boldsymbol{\theta}_d^{\text{test}}$) of test document d and compute perplexity for

[9] http://ipv6.nlsde.buaa.edu.cn/zuoyuan/.

predicting the second half. With regard to MetaLDA, we fixed the matrices Φ^{train} and Λ^{train} output from the training procedure. On the first half of test document d, we computed the Dirichlet prior $\boldsymbol{\alpha}_d^{\text{test}}$ with Λ^{train} and the labels $\boldsymbol{f}_d^{\text{test}}$ of test document d (See Step 2a), and then point-estimated $\boldsymbol{\theta}_d^{\text{test}}$. We ran all the models 5 times with different random number seeds and report the average scores and the standard deviations.

In testing, we may encounter words that never occur in the training documents (a.k.a., unseen words or out-of-vocabulary words). There are two strategies for handling unseen words for calculating perplexity on test documents: ignoring them or keeping them in computing the perplexity. Here we investigate both strategies:

5.4.1 Perplexity computed without unseen words

In this experiment, the perplexity is computed only on the words that appear in the training vocabulary. Here we used 80% documents in each dataset as the training set and the remaining 20% as the test set.

Tables 5 and 6 show[10] the average perplexity scores with standard deviations for all the models. Note that: (1) The scores on AN with 150 and 200 topics are not reported due to overfitting observed in all the compared models. (2) Given the size of NYT, the scores of 200 and 500 topics are reported. (3) The number of latent topics in LLDA must equal to the number of document labels. (4) For PLLDA, we varied the number of topics per label from 5 to 50 (2 and 5 topics on NYT). The total number of topics used by PPLDA is the product of the number of labels and the number of topics per label.

The results show that the proposed MetaLDA outperformed all the competitors in terms of perplexity on nearly all the datasets, showing the benefit of using both document and word meta-information. Specifically, we have the following remarks:

- By looking at the models using only the document-level meta-information, we can see the significant improvement of these models over LDA, which indicates that document labels can play an important role in guiding topic modelling. Although the performance of the two variants of MetaLDA with document labels and DMR is comparable, our models run much faster than DMR, which will be studied later in Sect. 5.8.
- It is interesting that PLLDA with 50 topics for each label has better perplexity than MetaLDA with 200 topics in the 20NG dataset. With the 20 unique labels, the actual number of topics in PLLDA is 1000. However, if 10 topics for each label in PLLDA are used, which is equivalent to 200 topics in MetaLDA, PLLDA is outperformed by MetaLDA significantly.
- At the word level, MetaLDA-def-wf performed the best among the models with word features only. Moreover, our model has a clear advantage in running speed (see Table 13). Furthermore, comparing MetaLDA-def-wf with MetaLDA-def-def and MetaLDA-0.1-wf with LDA, we can see using the word features indeed improved perplexity.
- The scores show that the improvement gained by MetaLDA over LDA on the short text datasets is larger than that on the regular text datasets. This is expected because meta-information serves as complementary information in MetaLDA and can have significant impact when the data is sparse.
- It can be observed that models usually gained improved perplexity, if the Dirichlet parameter α is sampled/optimised, in line with [33]. We further study this in Sect. 5.4.3.

[10] For GPU-DMM and PTM, perplexity is not evaluated because the inference code for unseen documents is not public available. The random number seeds used in the code of LLDA and PLLDA are pre-fixed in the package. So the standard deviations of the two models are not reported.

Table 5 Perplexity comparison on the regular text datasets

Dataset	Reuters				20NG				NYT	
#Topics	50	100	150	200	50	100	150	200	200	500
LDA	677 ± 1	634 ± 2	629 ± 1	631 ± 1	2147 ± 7	1930 ± 7	1820 ± 5	1762 ± 3	2293 ± 8	2154 ± 4
MetaLDA-def-def	648 ± 3	592 ± 2	559 ± 1	540 ± 1	2093 ± 6	1843 ± 7	1708 ± 5	1626 ± 4	2258 ± 9	2079 ± 8
DMR	640 ± 1	577 ± 1	544 ± 2	526 ± 2	2080 ± 8	1811 ± 8	1670 ± 4	1578 ± 1	**2231** ± 13	**2013**±6
MetaLDA-dl-0.01	649 ± 2	582 ± 2	551 ± 3	530 ± 2	2067 ± 9	1821 ± 7	1680 ± 5	1590 ± 1	**2219** ± 4	**2018** ± 4
MetaLDA-dl-def	642 ± 3	576 ± 3	543 ± 1	526 ± 1	2050 ± 4	1804 ± 6	1675 ± 8	1589 ± 2	**2230** ± 3	**2022** ± 5
LF-LDA	841 ± 4	787 ± 4	772 ± 3	771 ± 4	2855 ± 21	2576 ± 3	2433 ± 7	2326 ± 8	2831 ± 2	2700 ± 5
WF-LDA	659 ± 2	616 ± 2	615 ± 1	613 ± 1	2089 ± 7	1875 ± 2	1784 ± 2	1727 ± 3	2287 ± 6	2134 ± 6
MetaLDA-0.1-wf	659 ± 3	621 ± 1	619 ± 1	623 ± 1	2098 ± 7	1887 ± 8	1796 ± 8	1744 ± 4	2283 ± 4	2143 ± 2
MetaLDA-def-wf	643 ± 2	582 ± 4	552 ± 3	535 ± 1	2068 ± 6	1819 ± 1	1685 ± 7	1600 ± 3	2260 ± 7	2095 ± 6
MetaLDA	**633** ± 2	**568** ± 2	**536** ± 2	**517** ± 1	**2025** ± 12	**1781** ± 8	**1640** ± 5	**1551** ± 6	**2217** ± 6	**2020** ± 6
Dataset	Reuters				20NG				NYT	
#Topics per label	5	10	20	50	5	10	20	50	2	5
PLLDA	714	708	733	829	1997	1786	1605	**1482**	2839	2846
LLDA	834				2607				2948	

The best results are highlighted in boldface

Table 6 Perplexity comparison without unseen words on the short text datasets. The best results are highlighted in boldface

Dataset	WS				TMN				AN	
#Topics	50	100	150	200	50	100	150	200	50	100
LDA	961 ± 6	878 ± 8	869 ± 6	888 ± 5	1969 ± 14	1873 ± 6	1881 ± 9	1916 ± 4	406 ± 14	422 ± 12
MetaLDA-def-def	884 ± 10	733 ± 6	671 ± 6	625 ± 6	1800 ± 11	1578 ± 19	1469 ± 4	1422 ± 6	352 ± 16	336 ± 11
DMR	845 ± 7	683 ± 4	607 ± 1	562 ± 2	1750 ± 8	1506 ± 3	1391 ± 7	1323 ± 5	326 ± 6	**290** ± 5
MetaLDA-dl-0.01	840 ± 7	693 ± 6	618 ± 3	588 ± 4	1767 ± 11	1528 ± 10	1416 ± 7	1345 ± 13	321 ± 13	303 ± 8
MetaLDA-dl-def	832 ± 4	679 ± 5	622 ± 7	582 ± 5	1720 ± 7	1505 ± 16	1395 ± 11	1325 ± 12	319 ± 9	293 ± 7
LF-LDA	1164 ± 6	1039 ± 17	1019 ± 11	992 ± 6	2415 ± 35	2393 ± 11	2371 ± 10	2374 ± 14	482 ± 17	514 ± 19
WF-LDA	894 ± 6	839 ± 6	827 ± 10	842 ± 4	1853 ± 6	1766 ± 12	1830 ± 60	1854 ± 45	397 ± 5	410 ± 6
MetaLDA-0.1-wf	889 ± 6	832 ± 3	839 ± 2	853 ± 4	1865 ± 4	1784 ± 2	1799 ± 9	1831 ± 6	388 ± 3	410 ± 8
MetaLDA-def-wf	830 ± 6	688 ± 8	624 ± 5	584 ± 4	1730 ± 14	1504 ± 3	1402 ± 13	1342 ± 4	346 ± 15	332 ± 8
MetaLDA	**774** ± 9	**627** ± 6	**572** ± 3	**534** ± 4	**1657** ± 4	**1415** ± 16	**1304** ± 6	**1235** ± 6	**314** ± 9	**293** ± 9
Dataset	WS				TMN				AN	
#Topics per label	5	10	20	50	5	10	20	50	5	10
PLLDA	1060	886	735	642	2181	1863	1647	1456	440	525
LLDA	1543				2958				392	

- On the AN dataset, there is no statistically significant difference between MetaLDA and DMR. On NYT, a similar trend is observed: the improvement in the models with the document labels over LDA is obvious but not in the models with the word features. Given the number of the document labels (194 of AN and 545 of NYT), it is possible that the document labels already offer enough information and the word embeddings have little contribution in the two datasets.

5.4.2 Perplexity computed with unseen words

To test the hypothesis that the incorporation of meta-information in MetaLDA can significantly improve the modelling accuracy in the cases where the corpus is sparse, we varied the proportion of documents used in training from 20 to 80% and used the remaining for testing. It is natural that when the proportion is small, the number of unseen words in testing documents will be large. Instead of simply excluding the unseen words in the previous experiments, here we compute the perplexity with unseen words for LDA, DMR, WF-LDA and the proposed MetaLDA. For perplexity calculation, $\phi_{k,v}^{\text{test}}$ for each topic k and each token v in the test documents is needed. If v occurs in the training documents, $\phi_{k,v}^{\text{test}}$ can be directly obtained. While if v is unseen, $\phi_{k,v}^{\text{unseen}}$ can be estimated by the prior:

$$\frac{\beta_{k,v}^{\text{unseen}}}{n_{k,\cdot}^{\text{train}} + \beta_{k,\cdot}^{\text{train}} + \beta_{k,\cdot}^{\text{unseen}}}.$$

For LDA and DMR which do not use word features, $\beta_{k,v}^{\text{unseen}} = \beta_{k,v}^{\text{train}}$; For WF-LDA and MetaLDA which are with word features, $\beta_{k,v}^{\text{unseen}}$ is computed with the features of the unseen token. Following Step 1c, for MetaLDA, $\beta_{k,v}^{\text{unseen}} = \prod_{l'}^{L_{\text{word}}} \delta_{l',k}^{g_{v,l}^{\text{unseen}}}$.

Figure 4 shows the perplexity scores on Reuters, 20NG, TMN and WS with 200, 200, 100 and 50 topics, respectively. MetaLDA outperformed the other models significantly with a lower proportion of training documents and relatively higher proportion of unseen words. The gap between MetaLDA and the other three models increases while the training proportion decreases. It indicates that the meta-information helps MetaLDA to achieve better modelling accuracy on predicting unseen words.

5.4.3 Perplexity evaluation for using MetaLDA as a hyper-parameter sampling approach

We further study how MetaLDA performs in terms of perplexity when used as a hyper-parameter sampling approach without meta-information. The experimental settings are the same as the ones used in Sect. 5.4.1. Table 7 shows the results of different variants of MetaLDA on hyper-parameter sampling. We would like to point out that MetaLDA-0.1-asym is equivalent to MetaLDA-0.1-def, MetaLDA-asym-0.01 is equivalent to MetaLDA-def-0.01, and MetaLDA-asym-asym is equivalent to MetaLDA-def-def in Table 3. Here we use the former to make the comparison clear. We have the following observations:

- In general, the best perplexity score is derived with the use of both asymmetric α and asymmetric β.

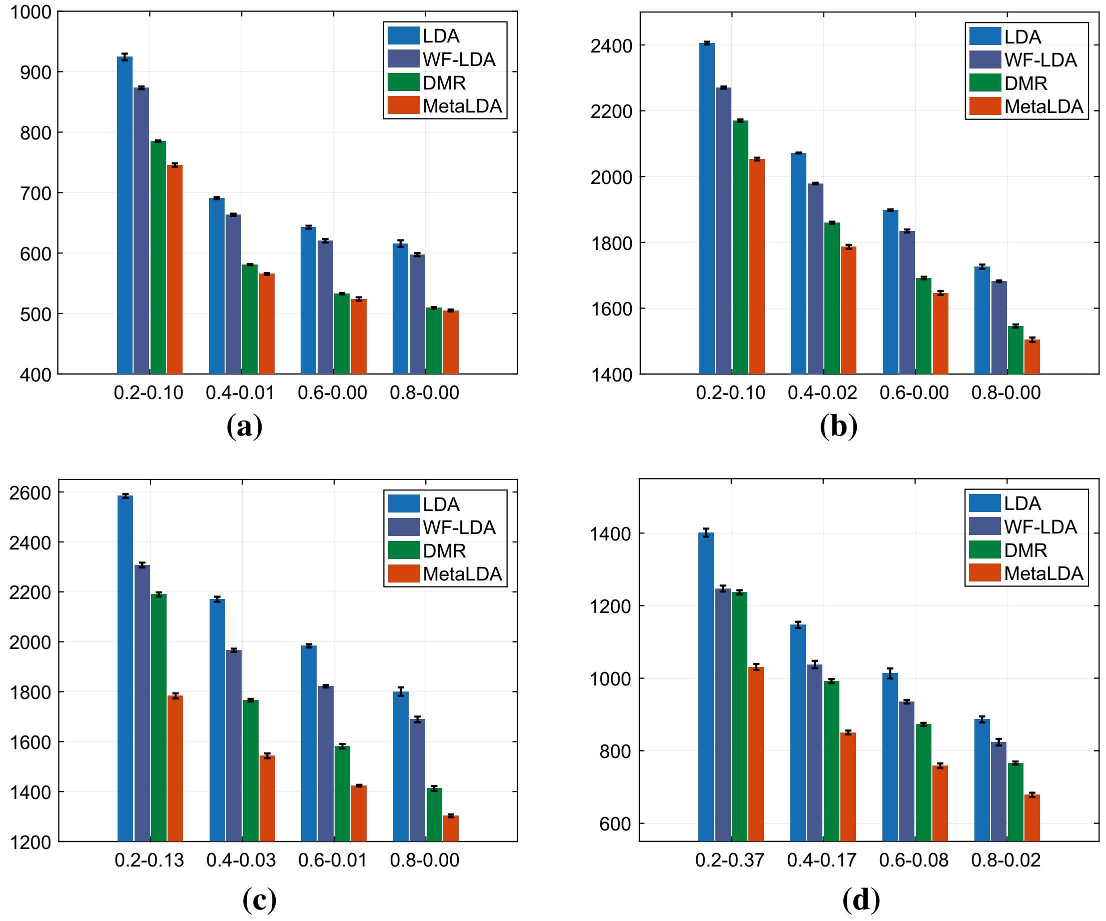


Fig. 4 Perplexity comparison with unseen words in different proportions of the training documents. Each pair of the numbers on the horizontal axis are the proportion of the training documents and the proportion of unseen tokens in the vocabulary of the test documents, respectively. For each setting, the four coloured bars from left to right correspond to LDA, WF-LDA, DMR and MetaLDA. The error bars are the standard deviations over 5 runs. **a** Reuters with 200 topics, **b** 20NG with 200 topics, **c** TMN with 100 topics, **d** WS with 50 topics

– If we fix the setting for the topic side and vary the setting for the document side (for example, compare MetaLDA-0.1-0.01, MetaLDA-sym-0.01 and MetaLDA-asym-0.01), we can derive that 1) the use of sampled priors (either symmetric or asymmetric) can significantly lower the perplexity scores, This is in line with the findings in [33]; 2) using asymmetric prior can further decrease perplexity.

– Similarly, fixing the setting for the document side and varying the setting for the topic side (for example, comparing MetaLDA-sym-0.01, MetaLDA-sym-sym and MetaLDA-sym-asym), we found that sampling either symmetric or asymmetric prior on per-topic word distributions does not significantly affect the perplexity scores, which also complies with [33]. However, there is a subtle difference: for our method an asymmetric prior on per-topic word distributions is marginally better, whereas it is often worse in [33].

– Now comparing the last row in Table 7 with the corresponding results in Tables 5 and 6 shows that constructing the priors with meta-information can further decrease the perplexity scores, which further proves our assumption that it is beneficial to use meta-information in topic modelling.

Table 7 Perplexity on 20NG with 200 topics, Reuters with 200 topics, WS with 100 topics

MetaLDA variants	20NG-200	Reuters-200	WS-100
MetaLDA-0.1-0.01 (LDA)	1762 ± 3	631 ± 1	878 ± 8
MetaLDA-0.1-sym	1774 ± 4	633 ± 1	888 ± 4
MetaLDA-0.1-asym	1764 ± 6	629 ± 2	884 ± 6
MetaLDA-sym-0.01	1652 ± 7	557 ± 5	744 ± 7
MetaLDA-sym-sym	1652 ± 6	557 ± 2	748 ± 6
MetaLDA-sym-asym	1641 ± 8	545 ± 2	743 ± 8
MetaLDA-asym-0.01	1618 ± 10	543 ± 1	726 ± 10
MetaLDA-asym-sym	1618 ± 11	542 ± 1	741 ± 11
MetaLDA-asym-asym	1626 ± 4	540 ± 1	733 ± 6

5.5 Topic coherence evaluation

We further evaluate the semantic coherence of the words in a topic learnt by LDA, PTM, DMR, LF-LDA, WF-LDA, GPU-DMM and MetaLDA. Here we use the normalised pointwise mutual information (NPMI) [1,16] to calculate topic coherence score for topic k with top T words:

$$\text{NPMI}(k) = \sum_{j=2}^{T}\sum_{i=1}^{j-1} \log \frac{p(w_j, w_i)}{p(w_j)p(w_i)} / -\log p(w_j, w_i),$$

where $p(w_i)$ is the probability of word i, and $p(w_i, w_j)$ is the joint probability of words i and j that co-occur together within a sliding window. Those probabilities were computed on an external large corpus, i.e., a 5.48 GB Wikipedia dump in our experiments. The NPMI score of each topic in the experiments is calculated with top 10 words ($T = 10$) by the Palmetto package.[11] Again, we report the average scores and the standard deviations over 5 random runs.

It is known that conventional topic models directly applied to short texts suffer from low quality topics, caused by the insufficient word co-occurrence information. Here we study whether or not the meta-information helps MetaLDA improve topic quality, compared with other topic models that can also handle short texts. Table 8 shows the NPMI scores on the three short text datasets. Higher scores indicate better topic coherence. All the models were trained with 100 topics. Besides the NPMI scores averaged over all the 100 topics, we also show the scores averaged over top 20 topics with highest NPMI, where “rubbish” topics are eliminated, following [37]. It is clear that MetaLDA performed significantly better than all the other models in WS and AN dataset in terms of NPMI, which indicates that MetaLDA can discover more meaningful topics with the document and word meta-information. We would like to point out that on the TMN dataset, even though the average score of MetaLDA is still the best, the score of MetaLDA overlaps with the others’ when allowing for standard deviation, which indicates the difference is not statistically significant.

[11] http://palmetto.aksw.org.

Table 8 Topic coherence (NPMI) on the short text datasets

	All 100 topics			Top 20 topics		
	WS	TMN	AN	WS	TMN	AN
LDA	−0.0030 ± 0.0047	0.0319 ± 0.0032	−0.0636 ± 0.0033	0.1025 ± 0.0067	0.137 ± 0.0043	−0.0010 ± 0.0052
PTM	−0.0029 ± 0.0048	0.0355 ± 0.0016	−0.0640 ± 0.0037	0.1033 ± 0.0081	0.1527 ± 0.0052	0.0004 ± 0.0037
DMR	0.0091 ± 0.0046	0.0396 ± 0.0044	−0.0457 ± 0.0024	0.1296 ± 0.0085	0.1472 ± 0.1507	0.0276 ± 0.0101
LF-LDA	0.0130 ± 0.0052	0.0397 ± 0.0026	−0.0523 ± 0.0023	0.1230 ± 0.0153	0.1456 ± 0.0087	0.0272 ± 0.0042
WF-LDA	0.0091 ± 0.0046	0.0390 ± 0.0051	−0.0457 ± 0.0024	0.1296 ± 0.0085	0.1507 ± 0.0055	0.0276 ± 0.0101
GPU-DMM	−0.0934 ± 0.0106	−0.0970 ± 0.0034	−0.0769 ± 0.0012	0.0836 ± 0.0105	0.0968 ± 0.0076	−0.0613 ± 0.0020
MetaLDA	**0.0311** ± 0.0038	**0.0451** ± 0.0034	**−0.0326** ± 0.0019	**0.1511** ± 0.0093	**0.1584** ± 0.0072	**0.0590** ± 0.0065

Table 9 Perplexity comparison for MetaLDA with different word embeddings on WS and TMN

Dataset	WS		TMN	
#Topics	50	100	50	100
GloVe-50	774 ± 9	627 ± 6	1657 ± 4	1415 ± 16
SkipGram-50	782 ± 11	643 ± 5	1678 ± 3	1449 ± 10
CBOW-50	781 ± 6	636 ± 9	1683 ± 11	1430 ± 6
GloVe-100	776 ± 3	648 ± 3	1653 ± 8	1418 ± 12
SkipGram-100	786 ± 14	651 ± 5	1685 ± 17	1444 ± 4
CBOW-100	778 ± 3	645 ± 7	1675 ± 11	1442 ± 16

Table 10 Topic coherence (NPMI) comparison for MetaLDA with different word embeddings on WS and TMN

	All 100 topics		Top 20 topics	
	WS	TMN	WS	TMN
GloVe-50	0.0311 ± 0.0038	0.0451 ± 0.0034	0.1511 ± 0.0093	0.1584 ± 0.0072
SkipGram-50	0.0251 ± 0.0052	0.0385 ± 0.0046	0.1405 ± 0.0081	0.1521 ± 0.0086
CBOW-50	0.0324 ± 0.0035	0.0430 ± 0.0048	0.1580 ± 0.0055	0.1532 ± 0.0027
GloVe-100	0.0286 ± 0.0043	0.0455 ± 0.0026	0.1473 ± 0.0082	0.1522 ± 0.0043
SkipGram-100	0.0277 ± 0.0041	0.0424 ± 0.0046	0.1508 ± 0.0058	0.1545 ± 0.0051
CBOW-100	0.0308 ± 0.0046	0.0408 ± 0.0035	0.1439 ± 0.0092	0.1505 ± 0.0102

5.6 Changing word embeddings

In the above experiments, we used the binarised 50-dimensional GloVe embeddings as word features to demonstrate the superiority of MetaLDA over all the other competitors. It is also interesting to study how the performance of MetaLDA changes while we use different word embeddings. In this set of experiments, we varied the sources (i.e., the methods used to train the word embeddings) as well as the dimensions of those word embeddings. Here we used the embeddings pre-trained by three methods: GloVe, SkipGram[12] and CBOW [20].[12] For each word embedding method, 50 and 100 dimensional embeddings were used.

Tables 9 and 10 show the perplexity and topic coherence performance of MetaLDA, respectively, on the WS and TMN datasets. We followed the experiment settings used in the previous sections, except for the word features. MetaLDA work marginally better with GloVe embeddings than with word2vec embeddings. However, the difference is not significant, given the standard errors. The reasons might be:

1. The binarisation could water down the differences between word embeddings. Therefore, minor differences in word embedding might not significantly influence the performance. But it is interesting to develop a model that can directly utilise the real-valued word embeddings.
2. Using the embeddings as the prior information could make MetaLDA insensitive to the quality of binarised word embeddings.

[12] http://vsmlib.readthedocs.io/en/latest/tutorial/getting_vectors.html.

Table 11 Top 5 related topics of the document labels in the WS dataset with 100 topics

Label	Topic number	Top 5 words	$\lambda_{l,k}$
Business	72	Exchange stock estate currency trading	12.11
	93	Trade capital export venture import	8.63
	94	Jobs marketing job stress advertising	7.99
	49	Bank financial banking finance insurance	7.06
	28	Business management services resources solutions	6.51
Computers	20	intel device digital apple chip	9.49
	66	Internet bandwidth speed connection test	6.57
	35	Computer software engineering architecture graphics	6.19
	48	Linux operating system unix library	5.10
	86	Memory computer virtual cache security	4.77
Culture&Arts&Entertainment	47	Art arts museum painting surrealism	11.16
	45	Guitar piano jazz orchestra instruments	6.87
	7	Religion ancient culture roman christian	6.41
	41	Album tom beatles band julia	6.32
	22	Culture American Chinese history Japanese	5.54
Education and science	68	Journal journals international conference research	7.36
	19	Theoretical models model reasoning framework	7.21
	81	Thesis dissertation technical empirical edu	7.04
	15	Physics quantum theory mechanics mathematics	6.40
	37	Research discovery scientific science scientists	5.77
Engineering	70	wheels car rims custom truck	5.95
	24	Electrical products equipment electric motor	5.80
	74	Car cars automobile models howstuffworks	5.68
	80	Automatic gear transmission China manual	4.84
	88	Engine diesel fuel cylinder turbine	4.72

Table 11 continued

Label	Topic number	Top 5 words	$\lambda_{l,k}$
Health	51	Diet calorie nutrition health energy	6.65
	96	HIV disease aids prevention heart	6.55
	98	Drug system respiratory effects drugs	5.89
	82	Physical therapy american therapists checkup	5.85
	52	Cancer lung tobacco smoking risk	5.69
Politics & Society	97	Cabinet prime minister appointment pbs	7.59
	18	System republic government parliamentary election	7.58
	83	Military revolution force navy army	7.27
	89	House gov congress legislation senate	5.21
	16	Democracy party democratic communist social	5.04
Sports	10	Football league rugby team stadium	11.21
	38	Tennis golf tournament woods volleyball	10.17
	27	Match cricket quarterfinal game playoff	8.45
	21	Tickets chicago bulls basketball boxing	6.68
	14	Soccer goalkeeper diego maradona kick	5.58

5.7 Qualitative analysis

Now we show that besides better quantitative performance, MetaLDA with meta-information also allows more informative and interesting interpretation of the discovered topics.

As discussed in Sect. 3, the latent variable $\lambda_{l,k}$ is the weight measuring the association between document label l and topic k. Each label can be interpreted as an unnormalised mixture of topics, represented by a K-dimensional vector $\boldsymbol{\lambda}_l$. Therefore, similar to finding the top words for each topic, ranking $\lambda_{l,k}$ can give us the most related topics for each label. Table 11 shows the top 5 related topics among 100 discovered by MetaLDA for the 9 document labels in the WS dataset. For each topic, the top 5 words (ranked with $\phi_{k,v}$) are listed. The results show that the topics are closely related to the labels. For example, the top 5 topics for the "Computers" category describe hardware, software, internet, and system, which are different aspects of computers. The "Sports" category broadly covers football, rugby, tennis, golf, cricket, etc. The major topics discussed in the "Health" related documents include diet, infectious diseases, lung cancer and its causes, and so on.

Table 12 Top 3 related labels of the topics in the WS dataset with 100 topics

Topic number	Top 5 words	Labels
46	Programming web java server code	Computers
		Education and science
		Engineering
54	Diet calorie nutrition health energy	Health
		Engineering
		Business
20	Intel device digital apple chip	Computers
		Culture&Arts&Entertainment
		Business
17	Movie fiction documentary film soundtrack	Culture&Arts&Entertainment
		Education and Science
		Sports

Furthermore, MetaLDA can also automatically assign the labels to the latent topics, which is known as automatic topic labelling [15]. The method proposed in [15] generates label from the top-ranked topic terms and the titles of Wikipedia articles containing these terms. It is an ad hoc process. In contrast, MetaLDA automatically learns the association between the document labels and the latent topics via the association matrix $\boldsymbol{\lambda}$. Specifically, for each topic k, we rank the labels according the weight $\lambda_{l,k}$, and then retrieve the most likely labels for each topic. Table 12 shows some examples derived one the WS dataset. For instance, topic 46 is about web programming. The most probable label for this topic assigned by MetaLDA is “Computers”. The second and third probable labels are also very related to this topic. Topic 17 is about movies, and the most probable label found by MetaLDA is “Culture&Arts&Entertainment”. It is clear that topics and their most probable labels are well correlated. All these findings demonstrate that MetaLDA is able to discover meaningful topics and label the topics automatically.

5.8 Running time

In this section, we empirically study the efficiency of the models in term of per-iteration running time. The implementation details of our MetaLDA are as follows:

- The SparseLDA framework [38] reduces the complexity of LDA to be sub-linear by breaking the conditional of LDA into three “buckets”, where the “smoothing only” bucket is cached for all the documents and the “document only” bucket is cached for all the tokens in a document. We adopted a similar strategy when implementing MetaLDA. When only the document meta-information is used, the Dirichlet parameters α for different documents in MetaLDA are different and asymmetric. Therefore, the “smoothing only” bucket has to be computed for each document, but we can cache it for all the tokens, which still gives us a considerable reduction in computing complexity. However, when the word meta-information is used, the SparseLDA framework no longer works in MetaLDA as the β parameters for each topic and each token are different.

Table 13 Running time (seconds per iteration) on 80% documents of each dataset

Dataset	Reuters				WS				NYT	
#Topics	50	100	150	200	50	100	150	200	200	500
LDA	0.0899	0.1023	0.1172	0.1156	0.0219	0.0283	0.0301	0.0351	0.7509	1.1400
PTM	4.9232	5.8885	7.2226	7.7670	1.1840	1.6375	1.8288	2.0030	–	–
DMR	0.6112	0.9237	1.2638	1.6066	0.4603	0.8549	1.2521	1.7173	13.7546	31.9571
MetaLDA-dl-0.01	0.1187	0.1387	0.1646	0.1868	0.0396	0.0587	0.0769	0.1121	2.4679	4.9928
LF-LDA	2.6895	5.3043	8.3429	11.4419	2.4920	6.0266	9.1245	11.5983	95.5295	328.0862
WF-LDA	1.0495	1.6025	3.0304	4.8783	1.8162	3.7802	6.1863	8.6599	14.0538	31.4438
GPU-DMM	0.4193	0.7190	1.0421	1.3229	0.1206	0.1855	0.2487	0.3118	–	–
MetaLDA-0.1-wf	0.2427	0.4274	0.6566	0.9683	0.1083	0.1811	0.2644	0.3579	4.6205	12.4177
MetaLDA	0.2833	0.5447	0.7222	1.0615	0.1232	0.2040	0.3282	0.4167	6.4644	16.9735

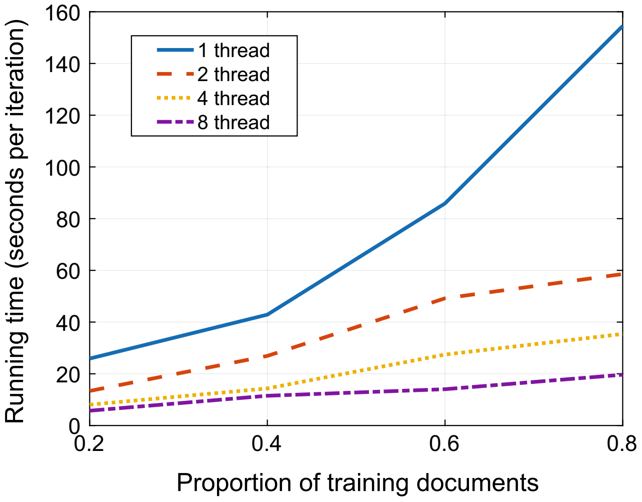


Fig. 5 MetaLDA's running time (seconds per iteration) on the NYT dataset with 500 topics with different proportions of training documents and different number of threads

- By adapting the Distributed framework in [25], our MetaLDA implementation runs in parallel with multiple threads, which makes MetaLDA able to handle larger document collections. The parallel implementation was tested on the NYT dataset.

The per-iteration running time of all the models is shown in Table 13. Note that:

- On the Reuters and WS datasets, all the models ran with a single thread on a desktop PC with a 3.40 GHz CPU and 16 GB RAM.
- Due to the size of NYT, we report the running time for the models that are able to run in parallel. All the parallelised models ran with 10 threads on a cluster with a 14-core 2.6 GHz CPU and 128 GB RAM.
- All the models were implemented in JAVA.
- As the models with meta-information add extra complexity to LDA, the per-iteration running time of LDA can be treated as the lower bound.

At the document level, both MetaLDA-df-0.01 and DMR use priors to incorporate the document meta-information and both of them were implemented in the SparseLDA framework. However, our variant is about 6 to 8 times faster than DMR on the Reuters dataset and more than 10 times faster on the WS dataset. Moreover, it can be seen that the larger the number of topics, the faster our variant is over DMR. At the word level, similar patterns can be observed: our MetaLDA-0.1-wf ran significantly faster than WF-LDA and LF-LDA especially when more topics are used (20–30 times faster on WS). It is not surprising that GPU-DMM has comparable running speed with our variant, because only one topic is allowed for each document in GPU-DMM. With both document and word meta-information, MetaLDA still ran several times faster than DMR, LF-LDA, and WF-LDA. On NYT with the parallel settings, MetaLDA maintains its efficiency advantage as well.

To further examine our model's scalability, we report the per-iteration running time of MetaLDA on NYT with 500 topics in Fig. 5. For this, we varied the proportion of training documents from 20 to 80% as well as the number of threads from 1 to 8. For the single thread version, when the training proportions change from 40 to 80% the per-iteration running time becomes 4 times slower. However, with multi-threading, our model scales much better. The per-iteration running time is only doubled while the training proportions quadruple. In terms of speed-up, the per-iteration running time increases nearly linearly with the number of threads. For example, given 60% training data, the per-iteration running time is reduced to half while the number of thread doubles.

6 Conclusion

In this article, we have presented a topic modelling framework named MetaLDA that can efficiently incorporate document and word meta-information. This results in a significant improvement over other models in terms of perplexity and topic quality. With two data augmentation techniques, MetaLDA enjoys full local conjugacy, allowing efficient Gibbs sampling, demonstrated by superiority in the per-iteration running time. MetaLDA[1] has been implemented within Mallet using the `DistributedLDA` framework, and works efficiently in a multicore context. Furthermore, without losing generality, MetaLDA can work with both regular texts and short texts. The improvement of MetaLDA over other models that also use meta-information is remarkable, particularly when the word-occurrence information is insufficient. Moreover, MetaLDA efficiently demonstrates that asymmetric-asymmetric LDA does beat regular symmetric LDA.

MetaLDA takes a particular approach for incorporating meta-information on topic models. However, the approach is general enough to be applied to other Bayesian probabilistic models that go beyond topics modelling, such as multi-label learning with sparse features [43]. Moreover, it would be interesting to extend our method to use real-valued meta-information directly without binarisation [41], which is the subject of future work.

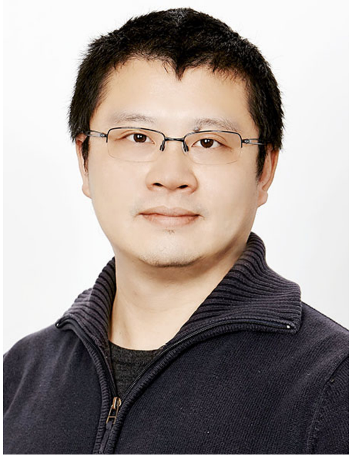
References

1. Aletras N, Stevenson M (2013) Evaluating topic coherence using distributional semantics. In: Proceedings of the 10th international conference on computational semantics, p 13–22
2. Andrzejewski D, Zhu X, Craven M (2009) Incorporating domain knowledge into topic modeling via Dirichlet forest priors. In: Proceedings of the 26th annual international conference on machine learning, p 25–32
3. Andrzejewski D, Zhu X, Craven M, Recht B (2011) A framework for incorporating general domain knowledge into Latent Dirichlet Allocation using first-order logic. In: Proceedings of the twenty-second international joint conference on artificial intelligence, p 1171–1177
4. Blei DM, Ng AY, Jordan MI (2003) Latent Dirichlet allocation. J Mach Learn Res 3:993–1022
5. Buntine W, Hutter M (2010) A Bayesian view of the Poisson–Dirichlet process. arXiv preprint arXiv:1007.0296
6. Buntine WL, Mishra S (2014) Experiments with non-parametric topic models. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining, p 881–890
7. Chen C, Du L, Buntine W (2011) Sampling table configurations for the hierarchical Poisson–Dirichlet process. In: Proceedings of the 2011 European conference on Machine learning and knowledge discovery in databases, p 296–311
8. Das R, Zaheer M, Dyer C (2015) Gaussian LDA for topic models with word embeddings. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing, p 795–804
9. Du L, Buntine W, Jin H, Chen C (2012) Sequential latent Dirichlet allocation. Knowl Inf Syst 31(3):475–503
10. Faruqui M, Tsvetkov Y, Yogatama D, Dyer C, Smith N (2015) Sparse overcomplete word vector representations. In: Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing, p 1491–1500
11. Guo J, Che W, Wang H, Liu T (2014) Revisiting embedding features for simple semi-supervised learning. In: Proceedings of the 2014 conference on empirical methods in natural language processing, p 110–120
12. Hong L, Davison BD (2010) Empirical study of topic modeling in Twitter. In: Proceedings of the first workshop on social media analytics, p 80–88
13. Hu C, Rai P, Carin L (2016) Non-negative matrix factorization for discrete data with hierarchical side-information. In: Proceedings of the 19th international conference on artificial intelligence and statistics, p 1124–1132
14. Kim D, Oh A (2017) Hierarchical Dirichlet scaling process. Mach Learn 106(3):387–418

15. Lau JH, Grieser K, Newman D, Baldwin T (2011) Automatic labelling of topic models. In: Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies, p 1536–1545
16. Lau JH, Newman D, Baldwin T (2014) Machine reading tea leaves: automatically evaluating topic coherence and topic model quality. In: Proceedings of the 14th conference of the european chapter of the association for computational linguistics, p 530–539
17. Li C, Wang H, Zhang Z, Sun A, Ma Z (2016) Topic modeling for short texts with auxiliary word embeddings. In: Proceedings of the 39th international ACM SIGIR conference on research and development in information retrieval, p 165–174
18. Mcauliffe JD, Blei DM (2008) Supervised topic models. Adv Neural Inf Process Syst 20:121–128
19. Mehrotra R, Sanner S, Buntine W, Xie L (2013) Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In: Proceedings of the 36th international ACM SIGIR conference on research and development in information retrieval, p 889–892
20. Mikolov T, Chen K, Corrado G, Dean J (2013) Efficient estimation of word representations in vector space. In: International conference on learning representations (workshop)
21. Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionally. Adv Neural Inf Process Syst 26:3111–3119
22. Miller GA (1995) WordNet: a lexical database for English. Commun ACM 38(11):39–41
23. Mimno D, McCallum A (2008) Topic models conditioned on arbitrary features with Dirichlet-multinomial regression. In: Proceedings of the 24th conference in uncertainty in artificial intelligence, p 411–418
24. Minka T (2000) Estimating a dirichlet distribution
25. Newman D, Asuncion A, Smyth P, Welling M (2009) Distributed algorithms for topic models. J Mach Learn Res 10:1801–1828
26. Nguyen DQ, Billingsley R, Du L, Johnson M (2015) Improving topic models with latent feature word representations. Trans Assoc Comput Linguist 3:299–313
27. Pennington J, Socher R, Manning C (2014) GloVe: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing, p 1532–1543
28. Petterson J, Buntine W, Narayanamurthy SM, Caetano TS, Smola AJ (2010) Word features for latent Dirichlet allocation. Adv Neural Inf Process Syst 23:1921–1929
29. Ramage D, Hall D, Nallapati R, Manning CD (2009) Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora. In: Proceedings of the 2009 conference on empirical methods in natural language processing, p 248–256
30. Ramage D, Manning CD, Dumais S (2011) Partially labeled topic models for interpretable text mining. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining, p 457–465
31. Teh YW, Jordan MI, Beal MJ, Blei DM (2006) Hierarchical Dirichlet processes. J Am Stat Assoc 101(476):1566–1581
32. Wallach HM (2008) Structured topic models for language. Ph.D. thesis, University of Cambridge
33. Wallach HM, Mimno DM, McCallum A (2009) Rethinking LDA: why priors matter. Adv Neural Inf Process Syst 22:1973–1981
34. Wang C, Blei DM (2011) Collaborative topic modeling for recommending scientific articles. In: Proceedings of the 17th ACM SIGKDD international conference on knowledge discovery and data mining, p 448–456
35. Xie P, Yang D, Xing E (2015) Incorporating word correlation knowledge into topic modeling. In: Proceedings of the 2015 conference of the north American chapter of the association for computational linguistics: human language technologies, p 725–734
36. Xun G, Gopalakrishnan V, Ma F, Li Y, Gao J, Zhang A (2016) Topic discovery for short texts using word embeddings. In: Proceedings of IEEE 16th international conference on data mining, p 1299–1304
37. Yang Y, Downey D, Boyd-Graber J (2015) Efficient methods for incorporating knowledge into topic models. In: Proceedings of the 2015 conference on empirical methods in natural language processing, p 308–317
38. Yao L, Mimno D, McCallum A (2009) Efficient methods for topic model inference on streaming document collections. In: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, p 937–946
39. Yin J, Wang J (2014) A Dirichlet multinomial mixture model-based approach for short text clustering. In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining, p 233–242
40. Zhao H, Du L, Buntine W (2017) Leveraging node attributes for incomplete relational data. In: Proceedings of the 34th international conference on machine learning, p 4072–4081

41. Zhao H, Du L, Buntine W (2017) A word embeddings informed focused topic model. In: Proceedings of the ninth Asian conference on machine learning, p 423–438
42. Zhao H, Du L, Buntine W, Liu G (2017) MetaLDA: a topic model that efficiently incorporates meta information. In: Proceedings of 2017 IEEE international conference on data mining, p 635–644
43. Zhao H, Rai P, Du L, Buntine W (2018) Bayesian multi-label learning with sparse features and labels, and label co-occurrences. In: Proceedings of the 21st international conference on artificial intelligence and statistics (**in press**)
44. Zhao WX, Jiang J, Weng J, He J, Lim EP, Yan H, Li X (2011) Comparing twitter and traditional media using topic models. In: Proceedings of the 33rd European conference on advances in information retrieval, p 338–349
45. Zhou M, Carin L (2015) Negative binomial process count and mixture modeling. IEEE Trans Pattern Anal Mach Intell 37(2):307–320
46. Zuo Y, Wu J, Zhang H, Lin H, Wang F, Xu K, Xiong H (2016) Topic modeling of short texts: a pseudo-document view. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, p 2105–2114

He Zhao is a Ph.D. student in the Faculty of Information Technology, Monash University, Australia. He received his B.Eng from Nankai University, China in 2011 and M.Eng from Nanjing University, China in 2014, respectively. His Ph.D. research project focuses on Bayesian (nonparametric) probabilistic models for discrete data (e.g., documents and networks).



Lan Du is currently a lecturer in the Faculty of Information Technology, Monash University. He previously worked in the language technology group in the Department of Computing, Macquarie university from 2012 to 2015. He received B.Sc. with honours and Ph.D. in computer science from the Australian National University (ANU) in 2017 and 2012, respectively. His research interests focus on statistical machine learning and its application in text analysis, relational learning, social network analysis, etc. With more than 30 papers published in these areas, he served/is serving on program committees for many top conferences in machine learning, data mining and natural language processing.

Wray Buntine is a full professor at Monash University in February 2014 after 7 years at NICTA in Canberra Australia. At Monash he is director of the Master of Data Science, the Faculty of IT's newest and in-demand degree, and was founding director of the innovative (online) Graduate Diploma of Data Science. He was previously at NICTA (Australia), Helsinki Institute for Information Technology, NASA Ames Research Center, University of California, Berkeley, and Google. He is known for his theoretical and applied work and in probabilistic methods for document and text analysis, social networks, data mining and machine learning. His recent focus has been with nonparametric methods in these areas.

Gang Liu Male, Ph.D., Associate Professor, Born in 1976. He got Ph.D. degree in Harbin Engineering University in China (2008), and major in computer applied technology. He conducted research in University of Illinois at Urbana–Champaign as visiting scholar in the group of Professor Jiawei Han in 2005. As a member of China Computer Federation, he has conducted and is conducting about 10 research projects such as National Science and Technology Support Plan and Chinese NSFC project as main researcher. He has published 30 papers in well-known journals such as JCIS, etc, which has been cited 20 times by SCI, EI. Dr. Liu has authored and co-authored 4 books in Chinese. He has filed 10 computer software copy authorities, and all of them have been authorized. He has developed and applied the advanced intelligent analysis and policy consistency verification technology, in auditing over 20 million attendees of Chinese social security.

Proceedings of Machine Learning Research 77:423–438, 2017 ACML 2017

A Word Embeddings Informed Focused Topic Model

He Zhao HE.ZHAO@MONASH.EDU
Lan Du LAN.DU@MONASH.EDU
Wray Buntine WRAY.BUNTINE@MONASH.EDU
Faculty of Information Technology, Monash University, Melbourne, Australia

Editors: Yung-Kyun Noh and Min-Ling Zhang

Abstract

In natural language processing and related fields, it has been shown that the word embeddings can successfully capture both the semantic and syntactic features of words. They can serve as complementary information to topics models, especially for the cases where word co-occurrence data is insufficient, such as with short texts. In this paper, we propose a focused topic model where how a topic focuses on words is informed by word embeddings. Our models is able to discover more informed and focused topics with more representative words, leading to better modelling accuracy and topic quality. With the data argumentation technique, we can derive an efficient Gibbs sampling algorithm that benefits from the fully local conjugacy of the model. We conduct extensive experiments on several real world datasets, which demonstrate that our model achieves comparable or improved performance in terms of both perplexity and topic coherence, particularly in handling short text data.

Keywords: Topic Models, Word Embeddings, Short Texts, Data Augmentation

1. Introduction

With the rapid growth of the internet, huge amounts of text data are generated everyday in social networks, online shopping and news websites, etc. Probabilistic topic models such as Latent Dirichlet Allocation (LDA) (Blei et al., 2003) are popular approaches for text analysis, by discovering latent topics from text collections.

Recently, word embeddings generated by GloVe (Pennington et al., 2014) and word2vec (Mikolov et al., 2013), have attracted a lot of attention in Natural Language Processing (NLP) and related fields. It has been shown that trained on large corpus, word embeddings can capture both the semantic and syntactic features of words so that similar words are close to each other in the embedding space. Therefore, if word embeddings can be used in topic models, it should improve modelling accuracy and topic quality. Moreover, as conventional topic models usually require a sufficient amount of word co-occurrences to learn meaningful topics, they can suffer from a large performance degradation over short texts (e.g., tweets and news headlines) because of insufficient word co-occurrence information. In such cases, the semantic and syntactic features of words encoded in embeddings can play a more important role, serving as complementary information.

On the other hand, sparse topics are preferred in topic modelling, which means most topics should be specific and are encouraged to focus on a small subset of the vocabulary. The topic sparsity is first implemented by using a small concentration parameter of the Dirichlet prior over topics (0.01 usually works well) (Wallach et al., 2009a). Recently, sparsity-enforcing priors such as the Indian Buffet Process compound Dirichlet (Williamson et al., 2010) have been proposed on topics (Wang and Blei, 2009), which place a "hard" constraint on the words that a topic can focus on. These sparsity-enforcing priors lead to compression as well as an easier interpretation of topics.

Inspired by those two lines of work, in this paper, we propose a Word Embedding Informed Focused Topic Model (WEI-FTM). This allows improved model accuracy and topic quality, especially for the cases where word co-occurrence data is poor, such as with short texts. Specifically, WEI-FTM applies a sparsity-enforcing prior on topics, allowing a subset of words to describe a topic. Unlike conventional FTM, where the focusing is learnt purely on the word co-occurrences, the focusing in WEI-FTM is additionally informed by external word embeddings. In this way, the proposed model encourages the topics to focus on the words that are more semantically or syntactically related, which is preferred in topic modelling.

WEI-FTM has the following key properties:

1. Compared to FTM, our model is able to discover more informed focused topics with more representative words, which leads to better model accuracy and topic quality.
2. Unlike most models incorporating word embeddings, our model does so by using them as prior knowledge, which we argue is a more coherent approach.
3. With the data augmentation technique, the inference of WEI-FTM can be done by an efficient and closed-form Gibbs sampling algorithm that benefits from local conjugacy of the model.
4. Finally, besides the word distribution of topics, our model also offers an alternative topic presentation over words, which can be obtained from word embeddings. It gives us a new way of interpreting topics and even better topic quality.

We conduct extensive experiments with several real datasets including both regular and short texts in various domains. The experimental results demonstrate that WEI-FTM achieves improved performance in terms of perplexity, topic coherence, and running time.

2. Related Work

In this section, we review three lines of related work: focused topic models, topic models with word embeddings, and short text topic models.

Focused Topic Models Focusing in topic models is first introduced on documents, allowing a document to focus on a subset of topics. Williamson et al. (2010) proposed the Focused Topic Model (FTM) on the document side with the Indian Buffet Process compound Dirichlet prior, where topics are selected by the IBP (Ghahramani and Griffiths, 2006). Zhou et al. (2012a) proposed a focused Poisson factorisation model with the negative binomial distribution, which can be viewed as a generalisation of FTM. Recently, Gan et al. (2015a) introduced a deep focused Poisson factorisation model. Instead of using IBP, document focusing in the model is constructed by stacking multiple layers of binary latent variables, connected by Gaussian weights. With the augmentation of the Pólya Gamma distribution (Polson et al., 2013), the model can be sampled with full conjugacy.

Unlike most of the previous approaches, the focussing in our model is applied to topics, not documents. The closest work to ours is the Sparse-Smooth Topic Model (Wang and Blei, 2009) which applied the IBP compound Dirichlet prior on topics, allowing a topic to focus on a subset of words. Teh and Gorur (2009) proposed the Stable-Beta IBP, a generalised IBP with a discount parameter. The Stable-Beta IBP can be used to model the power law behaviour in word occurrences. Furthermore, Archambeau et al. (2015) introduced the Latent IBP Dirichlet Allocation (LIDA), which uses the Stable-Beta IBP compound Dirichlet prior for both document focusing and topic focusing.

Topic Models with Word Embeddings Recently, there is growing interest in incorporating word features in topic models. For example, DF-LDA (Andrzejewski et al., 2009) incorporates word must-links and cannot-links using a Dirichlet forest prior in LDA; MRF-LDA (Xie et al., 2015) encodes word correlations in LDA with a Markov random field; WF-LDA (Petterson et al., 2010) extends LDA to model word features with the logistic-normal transform. As word embeddings have gained great success in NLP, they have been used as popular word features for topic models. LF-LDA (Nguyen et al., 2015) integrates word embeddings into LDA by replacing the topic-word Dirichlet multinomial component with a mixture of a Dirichlet multinomial component and a word embedding component. Instead of generating word types (tokens), Gaussian LDA (GLDA) (Das et al., 2015) directly generates word embeddings with the Gaussian distribution. MetaLDA (Zhao et al., 2017b) is a topic model that incorporates both document and word meta information. However, in MetaLDA, word embeddings have to be binarised, which will lose useful information. Despite the exciting applications of the above models, their inference is usually less efficient due to the non-conjugacy and/or complicated model structures. Moreover, to our knowledge, most of the existing models with word embeddings are extensions of a full LDA model, and neither use the embeddings as information for the prior, like WF-LDA, nor do they use the embeddings with topic focusing.

Short Text Topic Models Analysis of short text with topic models has been an active area with the development of social networks. One popular approach is to aggregate short texts into larger groups, for example, Hong and Davison (2010) aggregates tweets by the corresponding authors and Mehrotra et al. (2013) shows that aggregating tweets by their hashtags yields superior performance over other aggregation methods. Recently, PTM (Zuo et al., 2016) aggregates short texts into latent pseudo documents. Another approach is to assume one topic per short document, known as mixture of unigrams or Dirichlet Multinomial Mixture (DMM) such as Yin and Wang (2014); Xun et al. (2016). Closely related to ours are short text models that use word feature like embeddings. For example, Xun et al. (2016) introduced an extension of GLDA on short texts which samples an indicator variable that chooses to generate either the type of a word or the embedding of a word and GPU-DMM (Andrzejewski et al., 2011) extends DMM with word correlations for short texts. Although existing models showed improved performance on short texts, there still exist some challenges. For aggregation-based models, it is usually hard to choose which meta information to use for aggregation. The "single topic" assumption makes DMM models lose the flexibility to capture different topic ingredients of a document. The incorporation of word embeddings in the existing models is usually less efficient.

3. Model Details

Now we introduce the details of the proposed model. In general, WEI-FTM is a focused topic model where the focusing of topics are learnt from the target corpus and informed by external word embeddings. Specifically, suppose a collection of D documents with a vocabulary of V tokens is denoted as $\mathcal{D}$ and the L dimensional embeddings of the tokens are stored in a matrix $\mathbf{F} \in \mathbb{R}^{V \times L}$. Similar to LDA, WEI-FTM generates document $d \in \{1, \cdots, D\}$ with a mixture of K topics. Unlike LDA, where a topic is a distribution over all the tokens in the vocabulary, WEI-FTM allows a topic $k \in \{1, \cdots, K\}$ to focus on fewer tokens. We introduce a binary matrix $\mathbf{B} \in \{0,1\}^{K \times V}$ where $b_{k,v}$ indicates whether topic k focuses on token v. Given $\boldsymbol{b_{k,:}}$, topic k is a distribution over a subset of the tokens, drawn from the Dirichlet distribution:

$$\boldsymbol{\phi_k} | \boldsymbol{b_{k,:}} \sim \text{Dirichlet}_V(\beta_0 \boldsymbol{b_{k,:}}) \tag{1}$$

where $\phi_{k,v} = 0$ iff $b_{k,v} = 0$.

To get informed by word embeddings, $b_{k,v}$ is drawn from the Bernoulli distribution whose parameter is constructed with word v's embeddings $\boldsymbol{f_{v,:}}$:

$$b_{k,v} \sim \text{Bernoulli}\left(\sigma(\pi_{k,v})\right) \tag{2}$$

$$\pi_{k,v} = \boldsymbol{f_{v,:}} \boldsymbol{\lambda_{k,:}}^T + c_k \tag{3}$$

where $\boldsymbol{\Lambda} \in \mathbb{R}^{K \times L}$, $\boldsymbol{c} \in \mathbb{R}^K$, and $\sigma(x) = \frac{1}{1+e^{-x}}$ is the logistic function.

If we view $\boldsymbol{\lambda_{k,:}}$ as the embeddings of topic k, the intuition of our model is that if the closer the semantic/syntactic meanings (encoded in the embeddings) of tokens v to topic k, the larger the probability of k being described by v. Acting as the bias of topic k, c_k captures the information irrelevant to the embeddings. Gaussian prior is then used for both $\boldsymbol{\lambda_{k,:}}$ and $\boldsymbol{c}$:

$$\boldsymbol{\lambda_{k,:}}, \boldsymbol{c} \sim \mathcal{N}(\mathbf{0}, (\sigma_0)^2 \mathbf{I}) \tag{4}$$

where $(\sigma_0)^2$ is a hyper-parameter that controls the Gaussian variance.

Figure 1 shows the graphical model of WEI-FTM and the generative process is as follows:

1. For each topic k:
   (a) Draw $\boldsymbol{\lambda_{k,:}}$ according to Eq. (4)
   (b) For each token v: Draw $b_{k,v}$ according to Eq. (2)
   (c) Draw $\boldsymbol{\phi_{k,:}}$ according to Eq. (1)
2. For each document d:
   (a) Draw $\boldsymbol{\theta_{d,:}} \sim \text{Dirichlet}_K(\alpha_0 \mathbf{1}_K)$
   (b) For the i^{th} word $w_{d,i}$ in document d:
      i. Draw topic $z_{d,i} \sim \text{Categorical}_K(\boldsymbol{\theta_{d,:}})$
      ii. Draw word $w_{d,i} \sim \text{Categorical}_V(\boldsymbol{\phi}_{z_{d,i},:})$

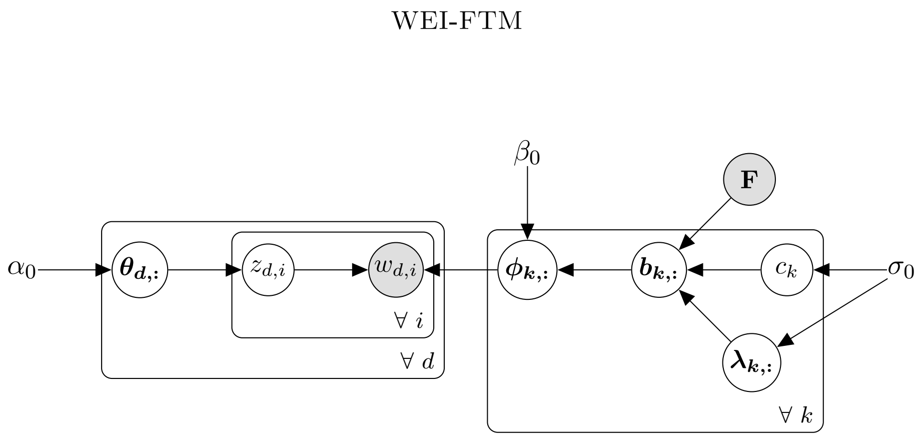


Figure 1: The graphical model of WEI-FTM. The i^{th} of N_d words in document d is $w_{d,i}$ and the corresponding topic is $z_{d,i}$. $\boldsymbol{\theta_{d,:}}$ is the topic distribution of d. α_0, β_0, and σ_0 are hyper-parameters of the model.

It is noteworthy that both $\boldsymbol{\pi_{k,:}} \in \mathbb{R}^V$ and $\boldsymbol{\phi_{k,:}}$ describe the weights of the words in topic k. Unlike $\phi_{k,v}$ which is obtained from the statistics of topic allocation of words, $\pi_{k,v}$ models the "similarity" of the embeddings of word v and topic k, obtained with the word embeddings. Therefore, π in our model can serve as an alternative presentation of topic k. This will be studied later in Section 5.3.

4. Inference

Unlike most existing models with word embeddings, our model facilitates the derivation of an efficient Gibbs sampling algorithm. With a data augmentation technique, WEI-FTM admits local conjugacy and a closed-form Gibbs sampling algorithm can be derived.

According to the generative process of WEI-FTM, the complete model likelihood is:

$$\prod_{d,i}^{D,N_d} p(w_{d,i}|z_{d,i}, \boldsymbol{\phi_{z_{i,n},:}})p(z_{i,n}|\boldsymbol{\theta_{d,:}}) \cdot \prod_{d}^{D} p(\boldsymbol{\theta_{d,:}}|\alpha_0) \cdot \prod_{k}^{K} p(\boldsymbol{\phi_{k,:}}|\boldsymbol{b_{k,:}}, \beta_0) \\ \cdot \prod_{k,v}^{K,V} p(b_{k,v}|\boldsymbol{\lambda_{k,:}}, c_k, \boldsymbol{f_{v,:}}) \cdot \prod_{k,l}^{K,L} p(\lambda_{k,l}|\sigma_0) \cdot \prod_{k}^{K} p(c_k|\sigma_0) \tag{5}$$

Sampling $z_{d,i}$ The sampling of a topic $z_{d,i}$ for a word $w_{d,i} = v$ is similar to LDA, while the candidate topics are limited to the topics that v describes:

$$p(z_{d,i} = k) \propto (\alpha_0 + m_{d,k}^{\neg i}) \frac{\beta_0 + n_{k,v}^{\neg d,i}}{\beta_0 V + n_{k,\cdot}^{\neg d,i}} \mathbb{I}_{(b_{k,v}=1)} \tag{6}$$

where $n_{k,v}^{\neg d,i} = \sum_{d',i'}^{D,N_{d'}} \mathbb{I}_{(d',i') \neq (d,i), w_{d',i'}=v, z_{d',i'}=k)}$, $m_{d,k}^{\neg i} = \sum_{i'}^{N_d} \mathbb{I}_{(i' \neq i, z_{d,i'}=k)}$, $n_{k,\cdot}^{\neg d,i} = \sum_{v}^{V} n_{k,v}^{\neg d,i}$, and $\mathbb{I}_{(\cdot)}$ is the indicator function.

Sampling $b_{k,v}$ Recall that $b_{k,v}$ indicates whether token v describes topic k. Therefore, if $n_{k,v} > 0$, which means there are words of v allocated to k, we do not need to sample $b_{k,v}$ (i.e., $p(b_{k,v}|n_{k,v} > 0) = 1$). When $n_{k,v} = 0$, the following Gibbs sampling for $b_{k,v}$ is used:

$$p(b_{k,v} = 1|n_{k,v} = 0) \propto \frac{\mathcal{B}(b_{k,\cdot}^{\neg v}\beta_0 + n_{k,\cdot}, \beta_0)}{\mathcal{B}(b_{k,\cdot}^{\neg v}\beta_0, \beta_0)} \sigma(\pi_{k,v}) \tag{7}$$

$$p(b_{k,v} = 0|n_{k,v} = 0) \propto 1 - \sigma(\pi_{k,v}) \tag{8}$$

where $\mathcal{B}(\cdot,\cdot)$ is the beta function and $b_{k,\cdot}^{\neg v} = \sum_{v'\neq v}^{V} b_{k,v'}$.

Sampling $\boldsymbol{\lambda_{k,:}}$ and $\boldsymbol{c}$ Recall that the likelihood of $b_{i,k}$ from Equation (2) is:

$$\frac{\left(e^{\pi_{k,v}}\right)^{b_{k,v}}}{1+e^{\pi_{k,v}}} \tag{9}$$

The above likelihood can be augmented by introducing an auxiliary variable: $\gamma_{k,v} \sim \mathrm{PG}(1,0)$ (Gan et al., 2015b), where PG denotes the Pólya Gamma distribution (Polson et al., 2013). The augmentation works as following:

$$\frac{\left(e^{\pi_{k,v}}\right)^{b_{k,v}}}{1+e^{\pi_{k,v}}} = \frac{1}{2} e^{(b_{k,v}-1/2)\pi_{k,v}} \int_0^\infty e^{-\gamma_{k,v}(\pi_{k,v})^2/2} p(\gamma_{k,v}) \mathrm{d}\gamma \tag{10}$$

Augmented in this way, the likelihood on $\pi_{k,v}$ has a Gaussian form, which means the likelihood of $\boldsymbol{\lambda_{k,:}}$ and $\boldsymbol{c}$ after the augmentation will be in Gaussian form as well. Given their Gaussian prior, one can sample $\boldsymbol{\lambda_{k,:}}$ as:

$$\boldsymbol{\lambda_{k,:}} \sim \mathcal{N}(\boldsymbol{\mu_k}, \mathbf{\Sigma_k}) \tag{11}$$

$$\boldsymbol{\mu_k} = \mathbf{\Sigma_k} \left(\textstyle\sum_v^V (b_{k,v} - 1/2 - c_k \gamma_{k,v}) \boldsymbol{f_{v,:}}^T \right) \tag{12}$$

$$\mathbf{\Sigma_k} = \left(\textstyle\sum_v^V \gamma_{k,v} \boldsymbol{f_{v,:}}^T \boldsymbol{f_{v,:}} + (\sigma_0)^{-2} \mathbf{I} \right)^{-1} \tag{13}$$

Also, $\boldsymbol{c}$ can be sampled similarly. Note that the Cholesky factorization can be applied to $\mathbf{\Sigma_k}$ to reduce the sampling complexity of $\boldsymbol{\lambda_{k,:}}$.

Finally, according to (Polson et al., 2013), we can sample $\gamma_{k,v}$ from its Pólya Gamma posterior: $\gamma_{k,v} \sim \mathrm{PG}(1, \pi_{k,v})$. One can approximate samples from the Pólya Gamma distribution by using a truncated sum of Gamma variables (Zhou et al., 2012b). In practice, a truncation level of 20 works well, so the sampling will be efficient.

Hyper-parameter Sampling We use a Gamma prior on $\beta_0 \sim \mathrm{Gamma}(\mu_0, \nu_0)$. The likelihood of β_0 is:

$$\prod_k^K \frac{\Gamma(b_{k,\cdot}\beta_0)}{\Gamma(b_{k,\cdot}\beta_0 + n_{k,\cdot})} \prod_v^V \frac{\Gamma(\beta_0 + n_{k,v})}{\Gamma(\beta_0)}^{\mathbb{I}_{(b_{k,v}=1)}} \tag{14}$$

Two auxiliary variables are then introduced: $q_k \sim \mathrm{Beta}(b_{k,\cdot}\beta_0, n_{k,\cdot})$ and $t_{k,v} \sim \mathrm{CRP}(\beta_0, n_{k,v})$, which is the probability on the partition size of a Chinese Restaurant Process (Lemma 16 Buntine and Hutter, 2012) with β_0 and $n_{k,v}$ as the concentration and the number of customers respectively. The posterior becomes augmented as: $\prod_{k,v}^{K,V} (q_k)^{\beta_0} (\beta_0)^{t_{k,v}} \mathbb{I}_{b_{k,v}=1}$, which is conjugate to the Gamma prior of β_0 (Zhao et al., 2017b,a).

Similarly, α_0 can be sampled as well. However, as we are more interested in studying the word side, for fair comparison with other models, α_0 is not sampled in the experiments.

5. Experiments

In this section, we evaluate the proposed WEI-FTM against several recent advances including focus topic models, models with word embeddings, and short text topic models. The experiments were conducted on five real datasets including both regular and short texts. We report the performance in terms of perplexity, topic coherence, and running time per iteration. We also qualitatively compare the focusing and the topic quality of models.

5.1. Datasets, Compared Models, and Parameter Settings

In the experiments, two regular text datasets and three short text datasets were used:

- **Reuters** is extracted from the Reuters-21578 dataset[1] where documents without any labels are removed. There are 11,367 documents, the vocabulary size is 8,817, and the average document length is 73.
- **KOS** is obtained from the UCI Machine Learning Repository[2], which is used by Archambeau et al. (2015). It has 3,430 documents and the vocabulary size is 6,677. A document has 100 words on average.
- **WS**, Web Snippets, contains 12,237 web search snippets, used by Li et al. (2016). The vocabulary contains 10,052 tokens and there are 15 words in one snippet on average.
- **TMN**, Tag My News, consists of 32,597 English RSS news snippets from Tag My News, used by Nguyen et al. (2015). Each snippet contains a title and a short description. There are 13,370 tokens in the vocabulary and the average length of a snippet is 18.
- **Twitter**, is extracted in 2011 and 2012 microblog tracks at Text REtrieval Conference (TREC)[3], preprocessed in Yin and Wang (2014). It has 11,109 tweets in total. The vocabulary size is 6,344 and a tweet contains 21 words on average.

All the datasets were tokenised by Mallet[4] and we removed the words that exist in less than 5 documents and more than 95% documents. For word embeddings, we used the 50-dimensional GloVe word embeddings[5] pre-trained on Wikipedia for all the models that incorporate word embeddings. We further removed the words that are not in the vocabulary of GloVe embeddings in all the datasets. Note that besides word embeddings, our model can be used to incorporate other kinds of word features such as word correlations as well.

We evaluate the performance of the following models:

- **WEI-FTM**: The proposed model, the hyper-parameter σ_0 was set to 1.0 and β_0 was sampled according to Eq. (14). To comprehensively exam the effect of word embeddings, we further compare with WEI-FTM without any word embeddings, named "WEI-FTM-no", which only samples the bias $\boldsymbol{c}$. WEI-FTM and WEI-FTM-no were implemented in Matlab.
- **LDA** (Blei et al., 2003): the baseline model. A LDA variant with β_0 sampled according to Eq. (14) is also in comparison, named as "LDA-sym". LDA and LDA-sym were implemented in Matlab.

[1] http://www.daviddlewis.com/resources/testcollections/reuters21578/

[2] https://archive.ics.uci.edu/ml/datasets/bag+of+words

[3] http://trec.nist.gov/data/microblog.html

[4] http://mallet.cs.umass.edu

[5] https://nlp.stanford.edu/projects/glove/

- **WF-LDA**, Word Feature LDA (Petterson et al., 2010): a model that incorporates word features on the prior of ϕ. As no code is publicly available, we implemented it in Matlab, where the optimisation part was done by LBFGS with the default parameter settings. Following Mimno and McCallum (2008), the Gaussian variance was set to 10 for the default word feature and 0.05 for the other features. Note that we adopt the idea of WF-LDA for incorporating word embeddings while it was not originally proposed for that.

- **LF-LDA**, Latent Feature LDA (Nguyen et al., 2015): a model that incorporates word embeddings. The original implementation[6] was used. Following the paper, we used 1500 and 500 MCMC iterations for initialisation and sampling respectively and set λ to 0.6, and used the same word embeddings in WEI-FTM.

- **LIDA-topic**, Latent IBP Dirichlet Allocation (Archambeau et al., 2015) with topic focusing only. Reviewed in Section 2, LIDA applies the Stable-Beta IBP for both document and topic focusing. As our intent here is on topic focusing, the Stable-Beta IBP is only applied on the topic side. Unlike the proposed WEI-FTM, its focusing is not informed by external word features. Because the code of LIDA is not public available, we implemented it in Matlab, according to the paper. The MH sampling details of ξ are not given in the paper, we adopted LBFGS to optimise ξ in terms of the likelihood shown in Eq. (43) in the paper. β_0 was sampled by Eq. (14), the same as WEI-FTM. For the other sampling algorithms and settings, we followed the paper, where ζ was set to 0.25.

- **SSTM**, Sparse-Smooth Topic Model (Wang and Blei, 2009): a focused topic model that allows a topic to focus on a subset of words. Discussed by Archambeau et al. (2015), SSTM can be viewed as a special case of LIDA (by fixing ξ and ζ to 1.0 and 0.0 respectively).

- **GPU-DMM**, Generalized Pólya Urn DMM (Li et al., 2016): a model that incorporates word correlations. The original implementation[7] was used. The word correlations were generated from the distances of the word embeddings. Following the paper, we set the hyper-parameters μ and ϵ to 0.1 and 0.7 respectively, and the symmetric document Dirichlet prior to $50/K$.

- **PTM**, Pseudo document based Topic Model (Zuo et al., 2016): a model for short text analysis. The original implementation[8] was used. Following the paper, we set the number of pseudo documents to 1000 and λ to 0.1.

All the models, except where noted, the Dirichlet parameter of the document-topic distribution (α_0) and of the topic-word distribution (β_0) were set to 0.1 and 0.01 respectively. Our intent is to fairly compare just the topic-word aspect of the models, so we keep the document-topic aspect equivalent. Also, 2000 MCMC iterations were used to train the models.

[6] https://github.com/datquocnguyen/LFTM

[7] https://github.com/NobodyWHU/GPUDMM

[8] http://ipv6.nlsde.buaa.edu.cn/zuoyuan/

In summary, WEI-FTM (the proposed model), WF-LDA, LF-LDA, and GPU-DMM are models with word embeddings; LDA, LDA-sym, WEI-FTM-no, LIDA-topic, and SSTM are models without word embeddings; GPUDMM and PTM are models particularly for short texts.

5.2. Perplexity Evaluation

Perplexity is a measure that is widely used (Wallach et al., 2009b) to evaluate the modelling accuracy of topic models. The lower the score, the higher the modelling accuracy. To get unbiased perplexity, we randomly selected some documents in a dataset as the training set and the remaining as the test set. We first trained a topic model on the training set to get the word distributions of each topic k ($\boldsymbol{\phi}_{k,:}$). Each test document d was split into two halves containing every first and every second words respectively. We then fixed the topics and trained the models on the first half to get the topic proportions ($\boldsymbol{\theta}_{d,:}$) of test document d and computed perplexity for predicting the second half. We ran all the models 5 times with different random number seeds and report the average perplexity scores and the standard deviations. Note that GPU-DMM and PTM provided no code for inference on new documents so no corresponding perplexity results are given.

Table 1: Perplexity on regular texts. The best and second results are in boldface and underline respectively.

Dataset	Reuters			KOS		
#Topics	*50*	*100*	*200*	*50*	*100*	*200*
LDA	672±2	634±1	627±1	1488±4	1395±5	1315±2
LDA-sym	672±2	631±1	631±3	1461±4	1384±4	1327±4
LF-LDA	841±4	771±4	634±1	1707±16	1637±8	1636±10
WF-LDA	**651**±3	621±2	618±1	1426±10	1357±4	1306±3
SSTM	670±4	633±1	629±1	1462±4	1384±2	1324±4
LIDA-topic	671±1	638±3	-	1462±5	1385±4	1340±5
WEI-FTM-no	666±1	629±2	628±1	1445±3	1377±7	1322±1
WEI-FTM	656±4	**616**±2	**610**±3	**1416**±4	**1335**±2	**1284**±6

Tables 1 and 2 show the perplexities of the compared models[9]. The results indicate that the proposed WEI-FTM performed best on nearly all the datasets. In regular text datasets, it can be observed that WF-LDA was the second best, approaching WEI-FTM closely. However, our model had a clear win in short text datasets, especially on TMN and WS, which indicates that our incorporation of word embeddings is more effective than WF-LDA. Moreover, our model runs much faster than WF-LDA, which will be studied later in Section 5.5. Not informed by the word embeddings, WEI-FTM-no performed similarly to vanilla LDA. The comparison between WEI-FTM-no and WEI-FTM shows that the benefit of using the information encoded in the word embeddings. While using word embeddings, LF-LDA did not get better results than LDA in terms of perplexity, which is in line with

[9] The experiment of LIDA-topic on Reuters with 200 topics did not finish in a reasonable time due to the failure of optimising ξ with LBFGS.

Table 2: Perplexity on short texts. The best and second results are in boldface and underline respectively.

Dataset	WS		TMN		Twitter	
#Topics	*50*	*100*	*50*	*100*	*50*	*100*
LDA	957±6	875±4	1956±14	1855±14	580±2	497±2
LDA-sym	955±8	886±6	1951±8	1880±6	579±3	498±2
LF-LDA	1164±6	1039±17	2415±35	2393±11	849±16	685±6
WF-LDA	888±8	829±8	1881±9	1833±11	582±9	507±10
SSTM	956±8	881±4	1932±5	1858±10	578±6	496±5
LIDA-topic	964±7	882±6	1951±11	1875±15	578±1	488±2
WEI-FTM-no	948±5	877±6	1943±13	1874±12	573±2	497±2
WEI-FTM	**885**±11	**819**±8	**1845**±6	**1747**±12	**559**±5	**479**±5

the report in Fu et al. (2016). While introducing focusing as well, LIDA-topic and SSTM were not observed to have clear improvements in terms of perplexity.

5.3. Coherence Evaluation

To evaluate the coherence of the learnt topics, we used Normalised Pointwise Mutual Information (NPMI) (Lau et al., 2014) to calculate topic coherence score for topic k with top T words: $\mathrm{NPMI}(k) = \sum_{j=2}^{T} \sum_{i=1}^{j-1} \log \frac{p(w_j,w_i)}{p(w_j)p(w_i)} / -\log p(w_j, w_i)$, where $p(w_i)$ is the probability of word i, and $p(w_i, w_j)$ is the joint probability of words i and j that co-occur together within a sliding window. Those probabilities are computed on an external large corpus, i.e., a 5.48GB Wikipedia dump in our experiments. The NPMI score of each topic in the experiments is calculated with top 10 words ($T = 10$) by the Palmetto package[10]. Again, we report the average scores and the standard deviations over 5 random runs of all the models.

Table 3: NPMI averaged over all the 100 topics on short text datasets. The best and second results are in boldface and underline respectively.

Datasets	WS	TMN	Twitter
LDA	-0.0044±0.0028	0.0343±0.0026	-0.0110±0.0064
LF-LDA	0.0130±0.0052	0.0397±0.0026	0.0008±0.0026
WF-LDA	**0.0289**±0.0060	0.0463±0.0015	-0.0074±0.0033
SSTM	-0.0012±0.0064	0.0381±0.0023	-0.0065±0.0040
LIDA-topic	-0.0063±0.0027	0.0420±0.0021	-0.0042±0.0036
WEI-FTM-ϕ	0.0043±0.0038	0.0417±0.0036	-0.0096±0.0017
WEI-FTM-π	-0.0092±0.0074	**0.0567**±0.0081	**0.0392**±0.0083
GPU-DMM	-0.0934±0.0106	-0.0970±0.0034	-0.1458±0.0104
PTM	-0.0029±0.0048	0.0355±0.0016	-0.0078±0.0008

[10] http://palmetto.aksw.org

Table 4: NPMI averaged over the top 20 topics. The best and second results are in boldface and underline respectively.

Datasets	WS	TMN	Twitter
LDA	0.1175±0.0122	0.1462±0.0036	0.0923±0.0042
LF-LDA	0.1230±0.0153	0.1456±0.0087	0.0972±0.0024
WF-LDA	**0.1499**±0.0131	0.1390±0.0527	0.0881±0.0090
SSTM	0.1163±0.0168	0.1476±0.0020	0.1002±0.0059
LIDA-topic	0.1147±0.0048	0.1553±0.0010	0.0964±0.0022
WEI-FTM-ϕ	0.1271±0.0015	0.1536±0.0041	0.0893±0.0026
WEI-FTM-π	0.1298±0.0079	**0.1832**±0.0172	**0.1615**±0.0120
GPU-DMM	0.0836±0.0105	0.0968±0.0076	0.0367±0.0164
PTM	0.1033±0.0081	0.1527±0.0052	0.0882±0.0037

It is known that conventional topic models directly applied to short texts suffer from low quality topics, caused by the insufficient word co-occurrence information. Here we study whether the focusing informed by word embeddings helps WEI-FTM improve topic quality, compared with other topic models that can also handle short texts. Table 3 and 4 show the NPMI scores for the compared models trained with 100 topics on the three short text datasets. Higher scores indicate better topic coherence. Besides the NPMI scores averaged over all the 100 topics, we also show the scores averaged over the top 20 topics with highest NPMI (Table 4), where "rubbish" topics are eliminated, following Yang et al. (2015). Recall that in WEI-FTM, the top words of the topics can be obtained by ranking either ϕ (WEI-FTM-ϕ) or π (WEI-FTM-π). We report both of them here.

Shown in Tables 3 and 4, it can be seen that in TMN and Twitter, WEI-FTM-π outperformed the others significantly. It indicates that the word embeddings successfully inform our model to learn better topics. It is also noteworthy that WEI-FTM-ϕ still got better NPMI than other models except WF-LDA in WS and TMN in general, although the word embeddings do not directly affect the top word ranking with ϕ.

To qualitatively analyse the topic qualities, we show the top 10 words of the topics of WEI-FTM in Table 5. The words in each topic were ranked by ϕ and π. It can be seen that in general, the coherence of the words ranked by π is better than that ranked by ϕ, which is in line with the overall NPMI scores shown in Table 3 and 4.

5.4. Focusing Analysis

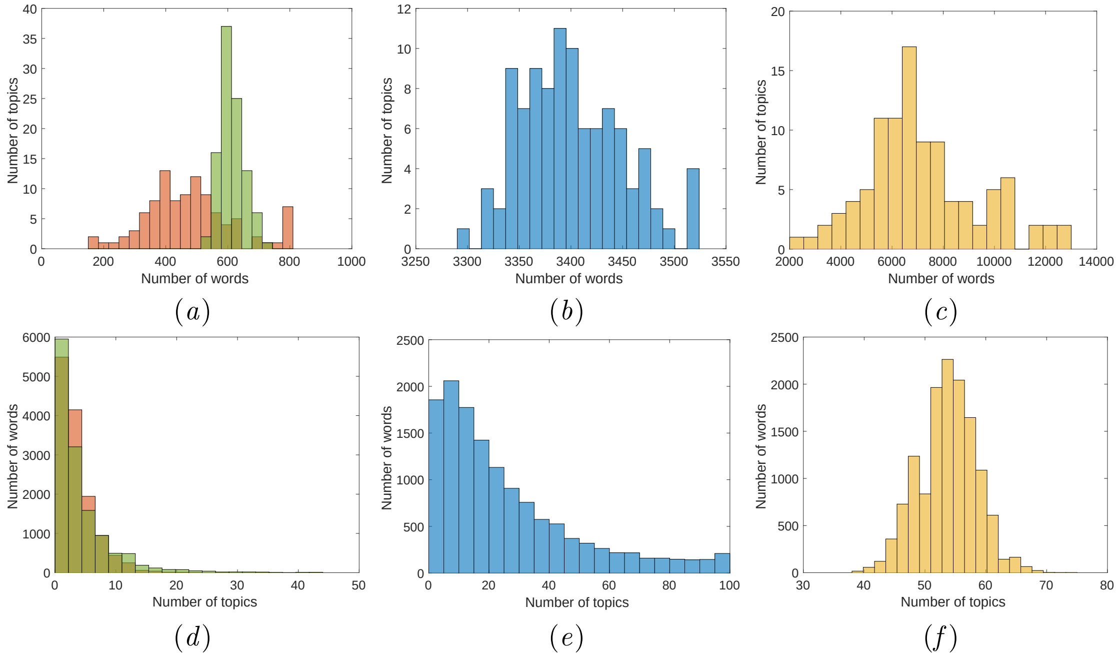
To compare the focusing in the focused topic models (WEI-FTM, WEI-FTM-no, SSTM, LIDA-topic), we show the histograms of the number of words per topic and the number of topics per word of two datasets in Figure 2 and 3. It can be observed that in WEI-FTM, the topics focused on fewer words than the others and the words described less topics. Compared to LIDA-topic, our model discovered more focused topics and the topics trend to be more diverse. It is also interesting to see how the word embeddings informed in the focusing: the topics in WEI-FTM-no are much less focused than those in WEI-FTM. This phenomenon helps explain why WEI-FTM gives better performance in the quantitative evaluations.

Table 5: Top 10 words of the topics discover by WEI-FTM on Twitter. Top 10 topics with the largest weights ($\sum_d^D \theta_{d,k}$) are selected. For each topic, the top words in the first row are ranked by ϕ and in the second are ranked by π respectively.

Topic	Top 10 words	NPMI
1	video bound kanye rogen west franco seth james kim kardashian	-0.0626
	starring movie sexy actress funny animated film comedy cartoon comedian	0.0680
2	china zone air japan east defense sea island disputed beijing	0.0141
	airspace diaoyu nato sovereignty territorial border resolution maritime military force	0.0543
3	watkins ian lostprophets guilty singer sex child baby rape pleaded	-0.0028
	murder convicted guilty sentence alleged rape imprisonment kidnapping trial sentenced	0.1291
4	swift taylor prince william bon jovi gala jon white winter	-0.0218
	sang concert sing singing princess dinner singer greeted danced tour	-0.0309
5	storm travel morning thanksgiving woman pill east plan winter weather	-0.0209
	weather mph rain crash sleet snow air jet storm coast	0.1036
6	scotland independence scottish paper white government independent salmond alex minister	0.0445
	parliament liberal prime election party democratic minister congress conservative vote	0.2352
7	nokia lumia phone window tablet smartphone inch device microsoft launch	0.1778
	playstation iphone ipad server smartphone gsm android xbox smartphones nintendo	0.1455
8	friday black thanksgiving shopping store day holiday retailer shopper year	0.0540
	dinner thanksgiving holiday shopping menu shop supermarket retail meal store	0.0491
9	patriot bronco manning brady peyton tom england sunday denver night	0.0389
	touchdown quarterback goalkeeper punt fumble defensive steelers coach nfl kickoff	0.1822
10	lakers bryant kobe los angeles washington extension year contract wizard	0.0770
	quarterback lakers nba coach knicks seahawks offseason postseason touchdown belichick	0.0520

Table 6: Running time (seconds per iteration) on 80% documents of each dataset

Dataset	Reuters		WS	
#Topics	*50*	*100*	*50*	*100*
LDA	6.6167	8.1239	1.1481	1.3904
LDA-sym	6.3883	8.2225	1.1255	1.3609
LF-LDA	2.6895	5.3043	2.4920	6.0266
WF-LDA	289.6488	636.8966	327.0750	724.7727
SSTM	10.7333	14.8040	4.0399	6.2999
LIDA-topic	23.4365	28.7910	3.9989	6.1942
WEI-FTM	24.6280	27.6666	6.4997	8.5074



(*a*) (*b*) (*c*)

(*d*) (*e*) (*f*)

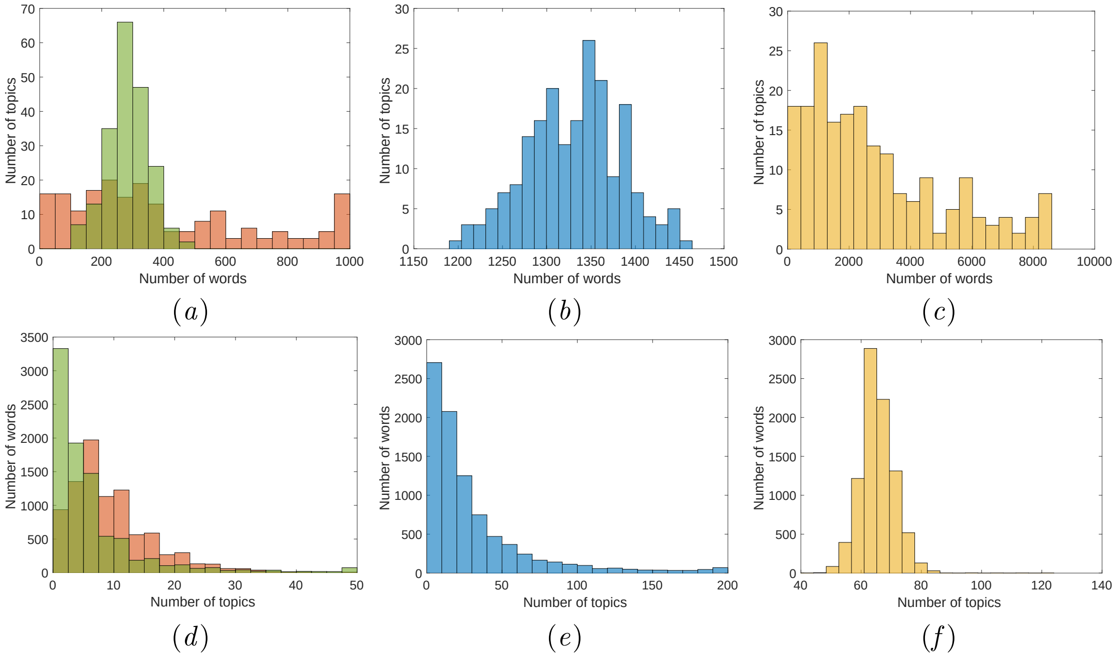
Figure 2: Histogram of the number of topics per word (a-c) and the number of words per topic (d-f) for the TMN dataset with 100 topics. Red: WEI-FTM, Green: LIDA-topic, Blue: SSTM, Yellow: WEI-FTM-no. The vocabulary size of TMN is 13,370. To show WEI-FTM and LIDA-topic in the same scale, we trimmed the topics and words with extremely low counts in (a).

5.5. Running Time

Here we empirically study the efficiency of the models. Table 5.5 shows the per-iteration running time of the compared models with different topics on the Reuters and WS datasets. Except LF-LDA (implemented in Java), the models were implemented in Matlab. All the models ran with the same settings on a cluster with a 14-core 2.6GHz CPU and 128GB RAM. It can be seen that although WF-LDA is comparable to WEI-FTM in some datasets, it runs much slower due to non-conjugacy.

6. Conclusion

In this paper, we have presented a focused topic model informed by word embeddings (WEI-FTM), which discovers more informed focused topics with more representative words, leading to better performance in terms of perplexity and topic quality. By leveraging the semantic and syntactic information encoded in word embeddings, our model is able to discover more focused and diverse topics with more representative words. In terms of inference, WEI-FTM enjoys full local conjugacy after augmentation, which facilitates an efficient Gibbs sampling algorithm for model inference. Without losing generality, WEI-FTM can work with both regular texts and short texts. The method of incorporating word



(a) (b) (c)

(d) (e) (f)

Figure 3: Histogram of the number of topics per word (a-c) and the number of words per topic (d-f) for the Reuters dataset with 200 topics. Red: WEI-FTM, Blue: SSTM, Green: LIDA-topic, Yellow: WEI-FTM-no. The vocabulary size of Reuters is 8,817. To show WEI-FTM and LIDA-topic in the same scale, we trimmed the topics and words with extremely low counts in (a).

embeddings introduced in this paper can also be applied to document features such as labels and authors, which is the subject of future work.

References

David Andrzejewski, Xiaojin Zhu, and Mark Craven. Incorporating domain knowledge into topic modeling via Dirichlet forest priors. In *ICML*, pages 25–32, 2009.

David Andrzejewski, Xiaojin Zhu, Mark Craven, and Benjamin Recht. A framework for incorporating general domain knowledge into Latent Dirichlet Allocation using first-order logic. In *IJCAI*, pages 1171–1177, 2011.

Cedric Archambeau, Balaji Lakshminarayanan, and Guillaume Bouchard. Latent IBP compound Dirichlet allocation. *TPAMI*, 37(2):321–333, 2015.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet Allocation. *JMLR*, 3 (Jan):993–1022, 2003.

W. Buntine and M. Hutter. A Bayesian view of the Poisson-Dirichlet process. *arXiv preprint arXiv:1007.0296v2 [math.ST]*, 2012.

Rajarshi Das, Manzil Zaheer, and Chris Dyer. Gaussian LDA for topic models with word embeddings. In *ACL*, pages 795–804, 2015.

Xianghua Fu, Ting Wang, Jing Li, Chong Yu, and Wangwang Liu. Improving distributed word representation and topic model by word-topic mixture model. In *ACML*, pages 190–205, 2016.

Zhe Gan, Changyou Chen, Ricardo Henao, David Carlson, and Lawrence Carin. Scalable deep Poisson factor analysis for topic modeling. In *ICML*, pages 1823–1832, 2015a.

Zhe Gan, R. Henao, D. Carlson, and Lawrence Carin. Learning deep sigmoid belief networks with data augmentation. In *AISTATS*, pages 268–276, 2015b.

Zoubin Ghahramani and T.L. Griffiths. Infinite latent feature models and the Indian buffet process. In *NIPS*, pages 475–482, 2006.

Liangjie Hong and Brian D Davison. Empirical study of topic modeling in twitter. In *Proc. of the First Workshop on Social Media Analytics*, pages 80–88, 2010.

Jey Han Lau, David Newman, and Timothy Baldwin. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *EACL*, pages 530–539, 2014.

Chenliang Li, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. Topic modeling for short texts with auxiliary word embeddings. In *SIGIR*, pages 165–174, 2016.

Rishabh Mehrotra, Scott Sanner, Wray Buntine, and Lexing Xie. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *SIGIR*, pages 889–892, 2013.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionally. In *NIPS*, pages 3111–3119, 2013.

David Mimno and Andrew McCallum. Topic models conditioned on arbitrary features with Dirichlet-multinomial regression. In *UAI*, pages 411–418, 2008.

Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. Improving topic models with latent feature word representations. *Transactions of the Association for Computational Linguistics*, 3:299–313, 2015.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

James Petterson, Wray Buntine, Shravan M Narayanamurthy, Tibério S Caetano, and Alex J Smola. Word features for Latent Dirichlet Allocation. In *NIPS*, pages 1921–1929, 2010.

Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American Statistical Association*, 108(504):1339–1349, 2013.

Yee W Teh and Dilan Gorur. Indian buffet processes with power-law behavior. In *NIPS*, pages 1838–1846, 2009.

Hanna M Wallach, David M Mimno, and Andrew McCallum. Rethinking LDA: Why priors matter. In *NIPS*, pages 1973–1981, 2009a.

H.M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation methods for topic models. In L. Bottou and M. Littman, editors, *ICML*, 2009b.

Chong Wang and David M Blei. Decoupling sparsity and smoothness in the discrete hierarchical Dirichlet process. In *NIPS*, pages 1982–1989, 2009.

Sinead Williamson, Chong Wang, Katherine A Heller, and David M Blei. The IBP compound Dirichlet process and its application to focused topic modeling. In *ICML*, pages 1151–1158, 2010.

Pengtao Xie, Diyi Yang, and Eric Xing. Incorporating word correlation knowledge into topic modeling. In *NAACL*, pages 725–734, 2015.

Guangxu Xun, Vishrawas Gopalakrishnan, Fenglong Ma, Yaliang Li, Jing Gao, and Aidong Zhang. Topic discovery for short texts using word embeddings. In *ICDM*, pages 1299–1304, 2016.

Yi Yang, Doug Downey, and Jordan Boyd-Graber. Efficient methods for incorporating knowledge into topic models. In *EMNLP*, pages 308–317, 2015.

Jianhua Yin and Jianyong Wang. A Dirichlet multinomial mixture model-based approach for short text clustering. In *SIGKDD*, pages 233–242. ACM, 2014.

He Zhao, Lan Du, and Wray Buntine. Leveraging node attributes for incomplete relational data. In *ICML*, pages 4072–4081, 2017a.

He Zhao, Lan Du, Wray Buntine, and Gang Liu. MetaLDA: a topic model that efficiently incorporates meta information. *arXiv preprint arXiv:1709.06365*, 2017b.

Mingyuan Zhou, Lauren Hannah, David B Dunson, and Lawrence Carin. Beta-negative binomial process and Poisson factor analysis. In *AISTATS*, pages 1462–1471, 2012a.

Mingyuan Zhou, Lingbo Li, David Dunson, and Lawrence Carin. Lognormal and Gamma mixed negative binomial regression. In *ICML*, volume 2012, page 1343, 2012b.

Yuan Zuo, Junjie Wu, Hui Zhang, Hao Lin, Fei Wang, Ke Xu, and Hui Xiong. Topic modeling of short texts: A pseudo-document view. In *SIGKDD*, pages 2105–2114, 2016.

Chapter 6

Interpretable Topic Structure Learning for Text Analysis

Topics discovered by BLFMs in text data are clusters of words, describing semantic concepts. Therefore, it is natural that topics are semantically correlated and such correlations can also be in structures. Learning the structures of topic correlations can be referred to as the topic structure learning problem.

Many existing models for topic structure learning impose hierarchical structures on top of document-topic distributions. While in the paper of Zhao et al. [2018b], a novel model for topic structure learning is proposed and it discovers topic hierarchies with structured latent variables imposed on topic-word distributions. The proposed structure has several advantages over previous ones, the most interesting one of which is that it can be combined with many other topic models with complex structures on document-topic distributions so that more interesting topic hierarchies can be discovered.

The following paper shows the details of this research:

- **H. Zhao**, L. Du, W. Buntine, M. Zhou, "Dirichlet Belief Networks for Topic Structure Learning", in *Neural Information Processing Systems* (**NeurIPS**) 2018.

On the other hand, in addition to correlations between topics, an individual topic is not semantically indivisible. For example, suppose a model discovers a topic about "entertainment" on a text corpus, which may consist of the words related to "music" and the words related to "sports." This can be because in the target corpus, the words of "music" and the words "sports" may co-occur a lot, so that the model cannot distinguish between them. In the paper of Zhao et al. [2018c], a novel model is proposed to discover such sub-topics by leveraging external word embeddings pre-trained on extremely large corpora. To my knowledge, this is the first work that discovers and solves the sub-topic problem in topic modelling. The following paper shows the details of this research:

- **H. Zhao**, L. Du, W. Buntine, M. Zhou, "Inter and Intra Topic Structure Learning with Word Embeddings", in *International Conference on Machine Learning* (**ICML**) 2018.

A potential future research direction is improving the scalability of the proposed models, which can be non-trivial. Specifically, to conduct batch training for BLFMs, a tricky challenge would be how to learn the global variables. However, the above two models impose complex hierarchical structures on the global variables, i.e., the topic-word distributions, which necessarily increases the complexity of designing a good learning scheme of learning the global variables. Therefore, how to efficiently training the above models requires further study.

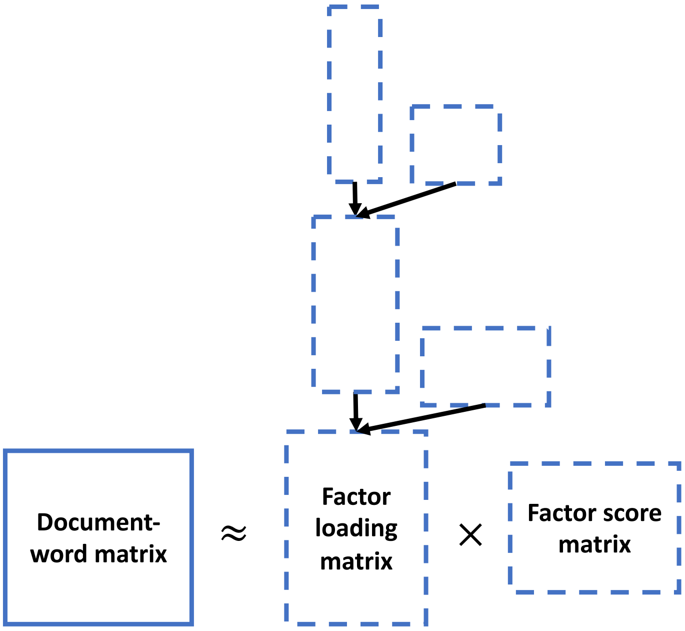


Figure 6.1: Model framework of Zhao et al. [2018b]. The blue rectangles with solid lines and dash lines are the data matrix (the document-word matrix containing word occurrences of the documents) and the latent matrices (the topic-word and doc-topic distributions) respectively. The topic-word distributions are hierarchical.

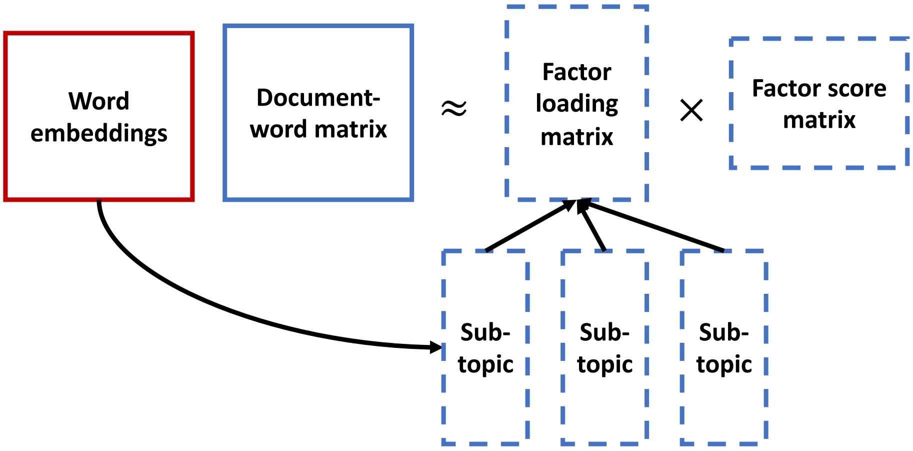


Figure 6.2: Model framework of Zhao et al. [2018c]. The blue rectangles with solid lines and dash lines are the data matrix (the document-word matrix containing word occurrences of the documents) and the latent matrices (the topic-word and doc-topic distributions), respectively. The red rectangles are the matrices of meta-data (the document meta-data and word embeddings).

The frameworks of the above two models are in Figure 6.1 and 6.2, respectively, which can be viewed as the hierarchical extensions of the basic framework of BLFMs shown in Figure 2.1 in Section 2.2.6 of Chapter 2. Specifically, the two models factorise the word occurrences matrix of a collection of documents into the factor loading matrix (the topic-word distributions) and the factor score matrix (the document-topic distributions). In the first model, the topic-word distributions are further factorised into the higher-layer topic-word distributions. While in the second model, the topic-word distributions are consisted of several sub-topics, the discovery of which is informed by the word embeddings.

The code of this research is released at `https://github.com/ethanhezhao/DirBN` and `https://github.com/ethanhezhao/WEDTM`.

Dirichlet belief networks for topic structure learning

He Zhao[1], **Lan Du**[1]*, **Wray Buntine**[1], **and Mingyuan Zhou**[2]*
[1]Faculty of Information Technology, Monash University, Australia
[2]McCombs School of Business, The University of Texas at Austin, USA

Abstract

Recently, considerable research effort has been devoted to developing deep architectures for topic models to learn topic structures. Although several deep models have been proposed to learn better topic proportions of documents, how to leverage the benefits of deep structures for learning word distributions of topics has not yet been rigorously studied. Here we propose a new multi-layer generative process on word distributions of topics, where each layer consists of a set of topics and each topic is drawn from a mixture of the topics of the layer above. As the topics in all layers can be directly interpreted by words, the proposed model is able to discover interpretable topic hierarchies. As a self-contained module, our model can be flexibly adapted to different kinds of topic models to improve their modelling accuracy and interpretability. Extensive experiments on text corpora demonstrate the advantages of the proposed model.

1 Introduction

Understanding text has been an important task in machine learning, natural language processing, and data mining. Text is discrete, unstructured, and often highly sparse. A popular way of analysing texts is to represent them as a set of latent factors via topic modelling or matrix factorisation. With great success in modelling text, probabilistic topic models discover a set of latent topics from a collection of documents. Those topics, as latent factors, can be interpreted by distributions over words and used to derive low dimensional representations of the documents. Specifically, most existing topic models are built on top of the following generative process: Each topic is a distribution over the words (i.e., *word distribution*, WD) in the vocabulary; each document is associated with a *topic proportion* (TP) vector; and a word in a document is generated by first drawing a topic according to the document's TP, then sampling the word according to the topic's WD.

In a Bayesian setting, TPs and WDs are both imposed on prior distributions. For example, one commonly-used prior for TP and WD is a Dirichlet distribution, as in Latent Dirichlet Allocation (LDA) (Blei et al., 2003). Recently, deep hierarchical priors, especially imposed on TPs, have been developed to generate hierarchical document representations as well as discover interpretable topic hierarchies. For example, there are hierarchical tree-structured constructions based on the Dirichlet Process (DP) or Chinese Restaurant Process (CRP), such as the nested CRP (nCRP) (Blei et al., 2010) and the nested hierarchical DP (Paisley et al., 2015); deep constructions based on restricted Boltzmann machines and neural networks such as the Replicated Softmax Model (RSM) (Hinton and Salakhutdinov, 2009), the Neural Autoregressive Density Estimator (NADE) (Larochelle and Lauly, 2012), and the Over-replicated Softmax Model (OSM) (Srivastava et al., 2013); models based on variational autoencoders (VAE) including Srivastava and Sutton (2017); Miao et al. (2017); Zhang et al. (2018). Recently, models that generalise the sigmoid belief network (Hinton et al., 2006) have been proposed, such as Deep Poisson Factor Analysis (DPFA) (Gan et al., 2015), Deep Exponential Families (DEF) (Ranganath et al., 2015), Deep Poisson Factor Modelling (DPFM) (Henao et al., 2015), and Gamma Belief Networks (GBNs) (Zhou et al., 2016).

*Corresponding authors

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

Compared with the considerable interest in deep models on TPs, to our knowledge, the counterparts on WDs have not been fully investigated. In this paper, we propose a new multi-layer generative process on WDs, as a self-contained module and an alternative to the single-layer Dirichlet prior. In the proposed model, WDs are the output units of the bottom layer in a DBN with hidden layers parameterised by Dirichlet-distributed hidden units and connected with gamma-distributed weights. Specifically, each Dirichlet unit in a hidden layer is a probability distribution over the words in the vocabulary and can be view as a "hidden" topic. In each layer, the Dirichlet prior of a topic is a mixture of the topics in the layer above. As the hidden units are drawn from Dirichlet, the proposed model is named the Dirichlet Belief Network, hereafter referred to as *DirBN*[2].

Compared with existing related deep models, DirBN has the following appealing properties: **1) Interpretability of hidden units**: Every hidden unit in every layer of DirBN is a probability distribution over the words, making them real topics that can be directly interpreted. **2) Discovering topic hierarchies**: The mixture structure of DirBN enables the model to enjoy a straightforward way of discovering semantic correlations of topics in two adjacent layers, which further form topic hierarchies with the multi-layer construction of the model. Due to the intrinsic abstraction effect of DBN, the topics in the higher layers are more abstract and can be treated as the generalisation of the ones in the lower layers. **3) Better modelling accuracy**: It is known that TPs are local variables (specific to individual document), while WDs are global variables over the target corpus. Unlike many other hierarchical parallels on TP, DirBN imposes a deep structure on WD, which "absorbs the information" from the entire corpus. It makes DirBN be able to get better modelling accuracy especially in the case of sparse texts such as tweets and news abstracts, where the context information of an individual document is not enough to learn a good model using existing approaches. **4) Adaptability**: As many sophisticated models on TPs usually use a simple Dirichlet prior on WDs, including the well-known ones such as Supervised Topic Model (Mcauliffe and Blei, 2008) and Author Topic Model (Rosen-Zvi et al., 2004), our DirBN can be easily adapted to them to further improve modelling accuracy and interpretability.

In conclusion, the contributions of this paper include: **1)** We propose DirBN, a deep structure that can be used as an advanced alternative to the Dirichlet prior on WDs with better modelling performance and interpretability. **2)** We demonstrate our model's adaptability by applying DirBN with several well-developed models, including Poisson Factor Analysis (PFA) (Zhou et al., 2012), MetaLDA (Zhao et al., 2017a), and GBN (Zhou et al., 2016). **3)** With proper data augmentation and marginalisation techniques, DirBN enjoys full local conjugacy, which facilitates the derivation of a simple and effective inference algorithm.

2 The proposed DirBN

In this section, we introduce the details of the generative and inference processes of DirBN.

2.1 Generative process

We first define the essential notation and review the basic framework of topic modelling, followed by the details of the proposed DirBN. Assume that the bag-of-words of document d in a corpus with N documents and V unique words in the vocabulary are stored in a count vector $\boldsymbol{x}_d \in \mathbb{N}_0^V$. A topic model with K topics is composed of the TP vector $\boldsymbol{\theta}_d \in \mathbb{R}_+^K$ for each document d and the WD vector $\boldsymbol{\phi}_k \in \mathbb{R}_+^V$ for each topic k ($k \in \{1, \cdots, K\}$). To generate a word in document d, one can first sample a topic according to its TP, and then sample the word type according to the topic's WD. Given this framework, many prior constructions of TPs have been proposed, such as the Dirichlet distribution in LDA, logistic normal distributions for modelling topic correlations in Correlated Topic Model (CTM) (Lafferty and Blei, 2006), nonparametric priors like the Hierarchical Dirichlet Process (Teh et al., 2012), and recently-proposed deep models like DPFA (Gan et al., 2015), DPFM (Henao et al., 2015), and GBN (Zhou et al., 2016). Unlike the extensive choices for constructing TP, the symmetric Dirichlet distribution on WDs still dominates in many advanced topic models. Here DirBN is a new hierarchical approach of constructing WDs, detailed as follows.

[2] Code available at `https://github.com/ethanhezhao/DirBN`

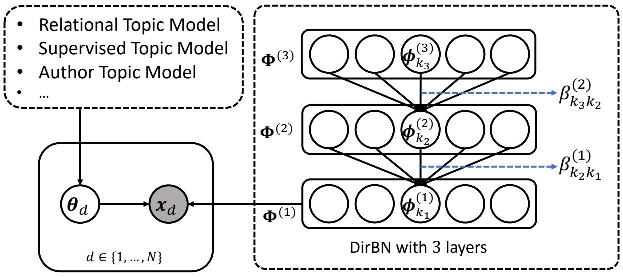


Figure 1: Demonstration of the generative process of DirBN with three layers.

A DirBN with T layers leaves the TPs of the basic framework untouched and draws $\boldsymbol{\phi}_k$ according to the following generative process:

$$\begin{aligned}
&\boldsymbol{\phi}_{k_T}^{(T)} \sim \mathrm{Dir}_V(\eta),\\
&\quad \cdots\\
&\boldsymbol{\phi}_{k_t}^{(t)} \sim \mathrm{Dir}_V(\boldsymbol{\psi}_{k_t}^{(t)}), \boldsymbol{\psi}_{k_t}^{(t)} = \sum_{k_{t+1}}^{K_{t+1}} \boldsymbol{\phi}_{k_{t+1}}^{(t+1)} \beta_{k_{t+1}k_t}^{(t)}, \beta_{k_{t+1}k_t}^{(t)} \sim \mathrm{Ga}(\gamma_{k_{t+1}}^{(t)}, 1/c^{(t)}),\\
&\quad \cdots\\
&\boldsymbol{\phi}_{k_1}^{(1)} \sim \mathrm{Dir}_V(\boldsymbol{\psi}_{k_1}^{(1)}),\ \boldsymbol{\psi}_{k_1}^{(1)} = \sum_{k_2}^{K_2} \boldsymbol{\phi}_{k_2}^{(2)} \beta_{k_2k_1}^{(1)}, \beta_{k_2k_1}^{(1)} \sim \mathrm{Ga}(\gamma_{k_2}^{(1)}, 1/c^{(1)}),
\end{aligned} \tag{1}$$

where 1) $\mathrm{Ga}(-,-)$ is the gamma distribution with shape and scale parameters and $\mathrm{Dir}_V(-)$ is the Dirichlet distribution[3]; 2) The superscript with a bracket over a variable indicates which layer it belongs to and $k_t \in \{1, \cdots, K_t\}$ is the topic index in the t-th layer; 3) The output of DirBN is $\boldsymbol{\phi}_{k_1}^{(1)}$, which corresponds to $\boldsymbol{\phi}_k$ in the basic framework and hereafter, we use $\boldsymbol{\phi}_{k_1}^{(1)}$ instead; 4) We further impose gamma priors on the following variables: $\eta \sim \mathrm{Ga}(a_0, 1/b_0)$, $\gamma_{k_{t+1}}^{(t)} \sim \mathrm{Ga}(\gamma_0^{(t)}/K_t, 1/c_0^{(t)})$, $\gamma_0^{(t)} \sim \mathrm{Ga}(e_0, f_0)$, $c_0^{(t)} \sim \mathrm{Ga}(g_0, 1/h_0)$, and $c^{(t)} \sim \mathrm{Ga}(g_0, 1/h_0)$. The generative process of a topic model equipped with DirBN is demonstrated in Figure 1.

The idea of our DirBN can be summarised as follows:

1. From a bottom-up view, DirBN is a multi-layer matrix factorisation, which factorises the matrix of the WDs in the t-th layer as: $\boldsymbol{\Phi}^{(t)} \sim \mathrm{Dir}(\boldsymbol{\Phi}^{(t+1)}\mathbf{B}^{(t)})$. Here we define $\boldsymbol{\Phi}^{(t)} \in \mathbb{R}_+^{V \times K_t}$ ($\boldsymbol{\phi}_{k_t}^{(t)}$ is the k_t-th column) and $\mathbf{B}^{(t)} \in \mathbb{R}_+^{K_{t+1} \times K_t}$ ($\boldsymbol{\beta}_{k_t}^{(t)}$ is the k_t-th column). From a top-down view, the model can be considered as a stochastic feedforward network (Tang and Salakhutdinov, 2013), where the input matrix in $\boldsymbol{\Phi}^{(T)}$, the output matrix is $\boldsymbol{\Phi}^{(1)}$, and the stochastic units are drawn from the Dirichlet distribution.
2. As DirBN is a Bayesian probabilistic model, consider a DirBN with only two layers as an example: each first-layer topic $\boldsymbol{\phi}_{k_1}^{(1)}$ is drawn from a Dirichlet with the topic-specific asymmetric parameter $\boldsymbol{\psi}_{k_1}^{(1)}$, which is a mixture of the second-layer topics. So the statistical strength is shared via the mixture, which plays an important role in handling sparse texts.
3. In DirBN, not only in the bottom layer, but also in any other layer t, each hidden unit is a distribution over the vocabulary and can be viewed as *real topic* directly interpreted by words. Although the bottom layer serves as the actual WDs for generating the words, the topics in the higher layers are involved with the belief prorogation in the network.
4. The weight $\beta_{k_{t+1}k_t}^{(t)}$ is drawn from a hierarchical gamma prior (i.e., the shape parameter $\gamma_{k_{t+1}}^{(t)}$ of the gamma prior on $\beta_{k_{t+1}k_t}^{(t)}$ is also drawn from a gamma). It allows topics in the $(t+1)$-th layer to contribute differently to those in the t-th layer. In addition, the hierarchical structure on $\beta_{k_{t+1}k_t}^{(t)}$ is similar to the one in Zhou (2015), which provides an

[3] $-$ can be a vector as a set of asymmetric parameters or a scalar as a symmetric parameter of Dirichlet

intrinsic shrinkage mechanism on $\boldsymbol{\beta}_{k_t}^{(t)}$. In other words, each k_t is expected to be sparsely connected by a subset of k_{t+1}. We will demonstrate the shrinkage effect of DirBN in the experiments.

2.2 Inference process

The learning of DirBN can be done by the inference of its latent variables, i.e., $\mathbf{\Phi}^{(t)}$ and $\mathbf{B}^{(t)}$ for all t. With several data augmentation techniques, we are able to derive a layer-wise Gibbs sampling algorithm facilitated by local conjugacy. Given θ and ϕ (despite their constructions), a topic model usually samples the topic assignment of each word in the corpus. After that, each topic k_1 is associated with a vector of word counts, denoted as $\boldsymbol{x}_{k_1}^{(1)} = [x_{1k_1}^{(1)}, \cdots, x_{Vk_1}^{(1)}]$, which encodes the semantic information of topic k_1 and is one of the *input count vectors* of DirBN in the inference process. Given the input vectors, the inference of DirBN involves two key steps: **1)** propagating the semantic information of the input vectors up to the top layer *via* latent counts; **2)** updating $\mathbf{\Phi}^{(t)}$ and $\mathbf{B}^{(t)}$ down to the bottom given the latent counts. Without loss of generality, we illustrate the inference details with a two-layer DirBN as follows[4]:

Propagating the latent counts from the bottom up By integrating $\boldsymbol{\phi}_{k_1}^{(1)}$ out from its multinomial likelihood, we can get the likelihood of $\boldsymbol{\psi}_{k_1}^{(1)}$ as:

$$\mathcal{L}\left(\boldsymbol{\psi}_{k_1}^{(1)}\right) \propto \frac{\Gamma(\psi_{\cdot k_1}^{(1)})}{\Gamma(\psi_{\cdot k_1}^{(1)} + x_{\cdot k_1}^{(1)})} \prod_{v}^{V} \frac{\Gamma(\psi_{vk_1}^{(1)} + x_{vk_1}^{(1)})}{\Gamma(\psi_{vk_1}^{(1)})}, \tag{2}$$

where $\Gamma(-)$ is the gamma function, $\psi_{\cdot k_1}^{(1)} = \sum_v^V \psi_{vk_1}^{(1)}$, and $x_{\cdot k_1}^{(1)} = \sum_v^V x_{vk_1}^{(1)}$. By integrating $\boldsymbol{\phi}_{k_1}^{(1)}$ out and introducing two auxiliary variables $q_{k_1}^{(1)}$ and $y_{vk_1}^{(1)}$, Eq. (2) can be augmented as (Zhao et al., 2017a):

$$\mathcal{L}\left(\boldsymbol{\psi}_{k_1}^{(1)}, q_{k_1}^{(1)}, y_{vk_1}^{(1)}\right) \propto \prod_{v}^{V} \left(q_{k_1}^{(1)}\right)^{\psi_{vk_1}^{(1)}} \left(\psi_{vk_1}^{(1)}\right)^{y_{vk_1}^{(1)}}, \tag{3}$$

where $q_{k_1}^{(1)} \sim \text{Beta}(\psi_{\cdot k_1}^{(1)}, x_{\cdot k_1}^{(1)})$ and $y_{vk_1}^{(1)} \sim \text{CRT}\left(x_{vk_1}^{(1)}, \psi_{vk_1}^{(1)}\right)$. Here CRT stands for the Chinese Restaurant Table distribution (Zhou and Carin, 2015; Zhao et al., 2017b). Now we can define $\boldsymbol{y}_{k_1}^{(1)} = [y_{1k_1}^{(1)}, \cdots, y_{Vk_1}^{(1)}]$, the *latent count vector* derived from the input count vector $\boldsymbol{x}_{k_1}^{(1)}$.

With $\psi_{vk_1}^{(1)} = \sum_{k_2}^{K_2} \phi_{vk_2}^{(2)} \beta_{k_2k_1}^{(1)}$, we can then distribute the latent count $y_{vk_1}^{(1)}$ on $\psi_{vk_1}^{(1)}$ to each second layer topic k_2 by:

$$\left(z_{v1k_1}^{(1)}, \cdots, z_{vK_2k_1}^{(1)}\right) \sim \text{Mult}\left(y_{vk_1}^{(1)}, \frac{\phi_{v1}^{(2)} \beta_{1k_1}^{(1)}}{\psi_{vk_1}^{(1)}}, \cdots, \frac{\phi_{vK_2}^{(2)} \beta_{K_2k_1}^{(1)}}{\psi_{vk_1}^{(1)}}\right), \tag{4}$$

where $z_{vk_2k_1}^{(1)}$ is the latent count allocated to k_2 and $\sum_{k_2}^{K_2} z_{vk_2k_1}^{(1)} = y_{vk_1}^{(1)}$.

We now note $\boldsymbol{x}_{k_2}^{(2)} = [x_{1k_2}^{(2)}, \cdots, x_{Vk_2}^{(2)}]$ where $x_{vk_2}^{(2)} = \sum_{k_1}^{K_1} z_{vk_2k_1}^{(1)}$. $\boldsymbol{x}_{k_2}^{(2)}$ can be viewed as one of the *output count vectors* of the first layer and also the input count vector of the second layer topic k_2.

In conclusion, to propagate the semantic information from the first to the second layer, we fist derive $y_{vk_1}^{(1)}$ from $x_{vk_1}^{(1)}$, then distribute $y_{vk_1}^{(1)}$ to all the second layer topics (i.e., $z_{vk_2k_1}^{(1)}$), and finally aggregate $z_{vk_2k_1}^{(1)}$ into $x_{vk_2}^{(2)}$.

Updating the latent variables from the top down After the latent counts are propagated, we start updating the latent variables from the top layer (i.e. the second layer here). Given $\boldsymbol{x}_{k_2}^{(2)}$, $\boldsymbol{\phi}_{k_2}^{(2)}$ is easy to sample from its Dirichlet posterior. With $z_{vk_2k_1}^{(1)}$ and $\sum_v^V \phi_{k_2,v}^{(2)} = 1$, we can sample $\beta_{k_2k_1}^{(1)}$ from its gamma posterior given the following likelihood:

$$\mathcal{L}\left(\beta_{k_2k_1}^{(1)}\right) \propto e^{-\beta_{k_2k_1}^{(1)}(-\log q_{k_1}^{(1)})} (\beta_{k_2k_1}^{(1)})^{z_{\cdot k_2k_1}^{(1)}}, \tag{5}$$

[4] Omitted details of inference as well as the overall algorithm are given in the supplementary materials.

where $z^{(1)}_{\cdot k_2 k_1} = \sum_v^V z^{(1)}_{vk_2k_1}$. Given the newly sampled $\boldsymbol{\phi}^{(2)}_{k_2}$ and $\beta^{(1)}_{k_2k_1}$, we can recompute $\boldsymbol{\psi}^{(1)}_{k_1}$ and sample $\boldsymbol{\phi}^{(1)}_{k_1}$ from its Dirichlet posterior. Now the inference of a two-layer DirBN is done.

3 Using DirBN in topic modelling

DirBN is a self-contained module on $\boldsymbol{\phi}$, leaving $\boldsymbol{\theta}$ untouched. Therefore, it can be used as an alternative to the simple Dirichlet prior on $\boldsymbol{\phi}$ in many existing models. The adaptability of DirBN enables us to easily apply it to advanced models so that those models can benefit from the advantages of DirBN. To demonstrate this, we adapt the proposed DirBN structure to the following models:

PFA+DirBN Poisson Factor Analysis (PFA) is a popular framework for topic analysis (DPFA (Gan et al., 2015), DPFM (Henao et al., 2015), GBN (Zhou et al., 2016) can be viewed as a deep extension to PFA). Specifically, we use the Bayesian nonparametric version of PFA named BGGPFA (Zhou et al., 2012), where $\boldsymbol{\theta}_d$ is constructed from a negative binomial process and $\boldsymbol{\phi}_k$ is drawn from a Dirichlet distribution. Note that there are close relationships between PFA and LDA, and between BGGPFA and HDP (Teh et al., 2012), analysed in Zhou (2018). Here we replace the Dirichlet construction on ϕ with DirBN, yielding a model named PFA+DirBN.

MetaLDA+DirBN MetaLDA (Zhao et al., 2017a, 2018a) is a supervised topic model that is able to incorporate document labels to inform the learning of $\boldsymbol{\theta}_d$. Keeping the structure on θ untouched, we replace the MetaLDA's structure on $\boldsymbol{\phi}$ with our DirBN to get a combined model that discovers the topic hierarchies informed by the document labels. The proposed model is able to discover the correlations between labels and topic hierarchies.

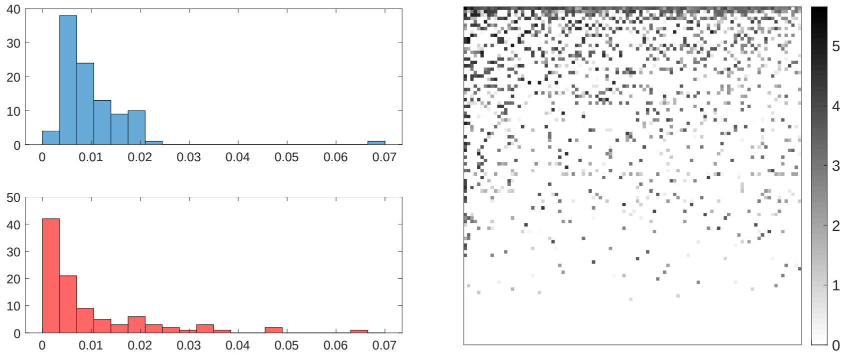
GBN+DirBN Recall that GBN (Zhou et al., 2015, 2016) imposes a hierarchical structure on θ, which is able to learn multi-layer document representations and topic hierarchies. Here we combine DirBN and GBN together to yield a "dual" deep model, where the GBN part is on θ and the DirBN part is on ϕ. Both parts discover topic hierarchies and the bottom-layer topics are shared by the two parts/hierarchies. It would be interesting to see how the two deep structures interact with each other.

4 Related work

As the proposed model introduces a hierarchical architecture on WDs (i.e., $\boldsymbol{\phi}$) in topic models, we first review various priors on $\boldsymbol{\phi}$, starting with the ones on sampling/optimising the Dirichlet parameters in topic models. The Dirichlet parameters in topic models were studied comprehensively in Wallach et al. (2009), which showed that Dirichlet with a symmetric parameter sampled from an uninformative gamma is the best choice. Actually, our DirBN can be reduced to this choice if $T = 1$ (i.e., DirBN-1, with one layer only). However, unlike the sampling/optimising approaches used in Wallach et al. (2009), DirBN-1 uses a negative binomial augmentation shown in Eq. (3), which leads to a simpler inference scheme. Recently, models like Zhao et al. (2017a,c, 2018b) construct informative and asymmetric Dirichlet priors by taking into account some external knowledge like word embeddings. Whereas DirBN learns the asymmetric priors purely based on the context of the target corpus.

Instead of Dirichlet, the Pitman-Yor process (PYP) has been used on WDs to model the power-law distribution of words, as in Sato and Nakagawa (2010); Buntine and Mishra (2014). Chen et al. (2015) used a transformed PYP prior on $\boldsymbol{\phi}$ to model multiple document collections. Lindsey et al. (2012) imposed a hierarchical PYP prior on $\boldsymbol{\phi}$ to discover word phrases. Besides PYP, the Indian Buffet Process (IBP) has been used as a prior on $\boldsymbol{\phi}$ to introduce word focusing on topics, as in Archambeau et al. (2015). In general, existing models use different priors on $\boldsymbol{\phi}$ for modelling various linguistic phenomena, which have different purposes to DirBN. The deep structures induced by DirBN on WDs have not yet been rigorously studied.

To our knowledge, most existing models explore the structure of topics by imposing a deep/hierarchical prior on $\boldsymbol{\theta}$. For example, hierarchical PYPs were used for domain adaptation in language models (Wood and Teh, 2009) and topic models (Du et al., 2012). nCRP (Blei et al., 2010) models topic hierarchies by introducing a tree-structured prior. Paisley et al. (2015); Kim et al. (2012); Ahmed et al. (2013) extended nCRP by either softening its constraints or applying it to different problems. Li and McCallum (2006) proposed the Pachinko Allocation model (PAM), which



(a) (b)

Figure 2: (a): Histograms of the normalised (latent) words counts. (b): $\mathbf{B}^{(1)}$.

captures the topic correlations with a directed acyclic graph. Recently, several deep extensions of PFA on θ have been proposed, including DPFA (Gan et al., 2015), DPFM (Henao et al., 2015), and GBN (Zhou et al., 2016). DPFM and GBN are the most related models to ours, which are also able to discover topic hierarchies. In DPFM and GBN, the higher-layer topics are not distributions over words but distributions over the topics in the layer below (they are called "meta-topics" in DPFM). To interpret those meta-topics, one needs to project them all the way down to the bottom-layer topics with matrix multiplication. Whereas in our model, the topics on all the layers are directly interpretable.

5 Experiments

The experiments were conducted on three real-world datasets, detailed as follows: **1)** Web Snippets (WS), containing 12,237 web search snippets labelled with 8 categories. The vocabulary contains 10,052 word types. **2)** Tag My News (TMN), consisting of 32,597 RSS news labelled with 7 categories. Each document contains a title and a description. There are 13,370 word types in the vocabulary. **3)** Twitter, extracted in 2011 and 2012 microblog tracks at Text REtrieval Conference (TREC)[5]. It has 11,109 tweets in total. The vocabulary size is 6,344.

With the framework of PFA, we compared three options of constructing ϕ: (1) The default setting of PFA, where ϕ is drawn from a symmetric Dirichlet distribution with parameter 0.05, i.e., $\boldsymbol{\phi}_k \sim \text{Dir}_V(0.05)$; (2) PFA+Mallet, where $\boldsymbol{\phi}_k \sim \text{Dir}_V(\alpha_0)$ and α_0 is sampled by Mallet [6]; (3) PFA+DirBN, the proposed model, where $\boldsymbol{\phi}_k$ is drawn from an asymmetric Dirichlet distribution specific to k, the parameter of which is constructed with the higher-layer topics. Note that Wallach et al. (2009) tested the option using specific asymmetric Dirichlet parameter, i.e., $\boldsymbol{\phi}_k \sim \text{Dir}_V([\alpha_1, \cdots, \alpha_V])$, but the performance is not as good as the symmetric parameter (the second one above). In addition, following a similar routine, we compared MetaLDA (Zhao et al., 2017a), and GBN (Zhou et al., 2016) with/without DirBN. Note that PFA is a widely used Bayesian topic model, MetaLDA is the state-of-the-art topic model capable of handling sparse texts, and GBN is reported Cong et al. (2017) to outperform many other deep models including DPFA (Gan et al., 2015), DPFM (Henao et al., 2015), nHDP (Paisley et al., 2015), and RSM (Hinton and Salakhutdinov, 2009).

For all the models, we ran 3,000 MCMC iterations with 1,500 burnin. For DirBN, we set $a_0 = b_0 = g_0 = h_0 = 1.0$ and $e_0 = f_0 = 0.01$. For PFA, MetaLDA, and GBN, we used their original implementations and settings, except that $\boldsymbol{\phi}$ is drawn from DirBN in the combined models. For all the models, the number of topics in each layer of DirBN was set to 100, i.e., $K_T = \cdots = K_1 = 100$. For GBN and GBN+DirBN, we set the number of topics in each layer of GBN to 100 as well. Due to the shrinkage mechanisms of PFA, GBN, and DirBN, the number of active topics will be adjusted according to the data. In all the experiments, we varied the number of layers of DirBN T from 1 to 3. For GBN+DirBN, the dual deep model, we fixed the number of layers of GBN as 3.

[5] http://trec.nist.gov/data/microblog.html

[6] http://mallet.cs.umass.edu

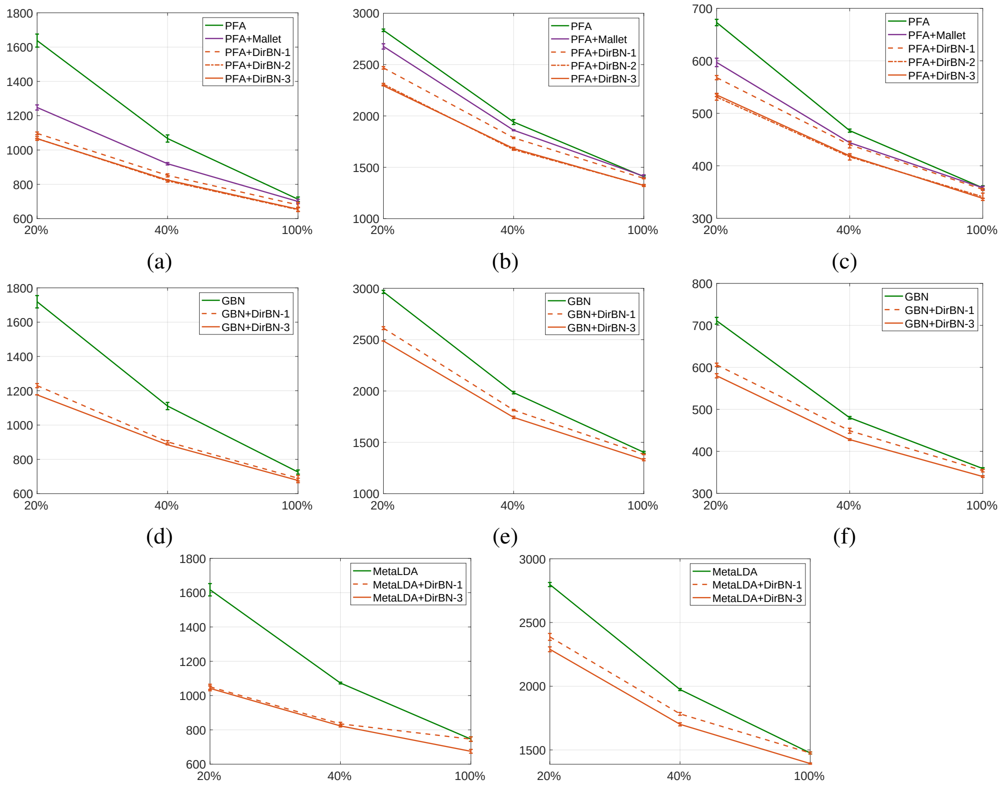


Figure 3: Perplexity (the vertical axis) with varied proportion (the horizontal axis) of the words for training in the training documents. (a-c): Results of the models based on PFA on WS, TMN, Twitter. (d-f): Results of the models based on GBN on WS, TMN, Twitter. (g,h): Results of the models based on MetaLDA on WS and TMN. The errorbars indicate the standard deviations of five runs. The number of a model indicates the number of layers used in DirBN. The results of MetaLDA and document classification on Twitter are not reported due to the unavailability of labels.

Demonstration of DirBN's shrinkage effect As previously discussed, DirBN has an intrinsic shrinkage mechanism that is able to automatically learn the number of active topics in each layer (i.e., the network width). We empirically demonstrate the shrinkage effect in Figure 2, with the results of PFA+DirBN-3 on the TMN dataset. Figure 2a plots the histograms of the normalised (latent) words counts $\sum_v x_{vk_t}^{(t)} / \sum_{vk_t} x_{vk_t}^{(t)}$ for all k_t where $x_{vk_t}^{(t)}$ is the word count for topic k_t. The blue and red bars are for the first- ($t = 1$) and the second-layer ($t = 2$) topics, respectively. The histogram indicates the number of topics (the vertical axis) that are with a specific word count (the horizontal axis). A topic with a larger word count is more important. The shrinkage effect is that large proportion of the topics are with very small word counts, indicating that the number of effective topics is less than the truncation (i.e., $K_t = 100$). This is more obvious, in the second layer. Moreover, we display $\log \mathbf{B}^{(1)}$ as an image in Figure 2b. The vertical and horizontal axes are for the second- and first-layer topics, respectively. We ranked the first- and second-layer topics by their word counts. The sparsity of $\mathbf{B}^{(1)}$ indicates that the first- and second-layer topics are sparsely connected. This also demonstrates the shrinkage effect of the model.

Quantitative results We report the per-heldout-word perplexity and topic coherence results. To compute perplexity, we randomly selected 80% of the documents in each dataset to train the models and 20% for testing. For each testing document, we randomly used one half of its words to infer its TP, and the other half to calculate perplexity. Topic coherence measures the semantic coherence in the most significant words (top words) of a topic. Here we used the Normalized Pointwise Mutual Information (NPMI) (Aletras and Stevenson, 2013; Lau et al., 2014) to calculate topic coherence

Table 1: Topic coherence with varied proportion of the words for training in the training documents. ± indicates the standard deviation of five runs. The best result in each column is in boldface.

	WS			TMN			Twitter		
Training words	20%	40%	100%	20%	40%	100%	20%	40%	100%
PFA	-0.070±0.010	0.008±0.002	0.062±0.011	-0.059±0.008	0.064±0.009	0.103±0.006	-0.003±0.003	0.031±0.003	0.046±0.002
PFA+Mallet	0.008±0.004	0.049±0.005	0.063±0.003	0.035±0.006	0.083±0.005	0.108±0.005	0.022±0.003	0.037±0.002	0.045±0.003
PFA+DirBN-1	0.013±0.003	0.052±0.004	0.060±0.006	0.031±0.003	0.080±0.001	0.108±0.008	0.019±0.004	0.037±0.004	0.049±0.007
PFA+DirBN-3	**0.021**±0.005	0.059±0.002	0.068±0.004	0.046±0.003	0.090±0.003	0.111±0.004	0.024±0.001	0.038±0.002	0.049±0.002
GBN	-0.072±0.013	0.007±0.005	0.069±0.009	-0.065±0.008	0.063±0.006	0.106±0.004	-0.005±0.005	0.032±0.002	0.047±0.00
GBN+DirBN-1	0.015±0.005	0.057±0.002	0.069±0.005	0.032±0.002	0.086±0.002	0.112±0.007	0.021±0.004	0.040±0.005	0.050±0.005
GBN+DirBN-3	0.018±0.006	**0.061**±0.004	**0.075**±0.002	0.048±0.003	**0.094**±0.004	**0.113**±0.004	**0.025**±0.003	**0.040**±0.002	**0.051**±0.003

Table 2: Topic hierarchy comparison in GBN+DirBN. Each row in boldface is the top 10 words in a first-layer topic. Each of these topics is associated with three most correlated topics in the second layer of DirBN (left) and GBN (right), respectively. The number associated with a second-layer topic is its (normalised) link weight to the first-layer topic.

police arrested man charged woman authorities death year found accused			
0.13	case charges accused trial courtattorney investigation judge allegations criminal	0.38	police arrested man charged year accused found charges woman death
0.13	police official killing attack deaddeath army security man family	0.19	police prison man china years arrested charges charged year chinese
0.11	woman men drug suicide girl sexual death found human york	0.15	china police chinese bomb fire people blast city artist officials
heat miami james lebron game nba finals celtics bulls wade			
0.43	season team game play run night star series fans career	0.97	heat miami james game nba finals lebron bulls mavericks dallas
0.15	nba playoffs court brink seeds defeated berth seed opponent semifinals	0.00	trial rajaratnam insider trading fund hedge raj anthony galleon case
0.10	win victory beat lead winning top fourth loss straight beating	0.00	music album lady gaga justin star pop band rock tour
facebook google internet social twitter online web media site search			
0.18	phone plan video technology mobile devices computer tech ceo content	0.22	facebook social internet google online twitter chief executive media web
0.14	company million buy billion corp industry sales companies consumers products	0.19	court lawsuit case facebook judge social federal internet google online
0.12	government report country nation pressure official state move released public	0.18	facebook social internet google online twitter world web media site
study cancer drug risk heart patients women researchers disease people			
0.12	rising percent high higher economic increase low growth strong recovery	0.91	study cancer drug risk researchers heart people patients health women
0.12	reactions periods technique method declared important realized treatment peril scores	0.04	world war years family oil dies year energy women american
0.10	study experience finding recent security kids challenges millions report special	0.18	facebook social internet google online twitter world web media site
nuclear japan plant power radiation crisis japanese fukushima crippled tokyo			
0.17	government united states officials state report country group official agency	0.53	nuclear japan plant power radiation crisis japanese fukushima earthquake tokyo
0.14	safety water nearby land found caused sea believed center parts	0.44	nuclear japan plant power radiation crisis japanese fukushima water tokyo
0.13	work plans part future system rules program bring offers decision	0.01	theater review broadway play york musical stage life time love

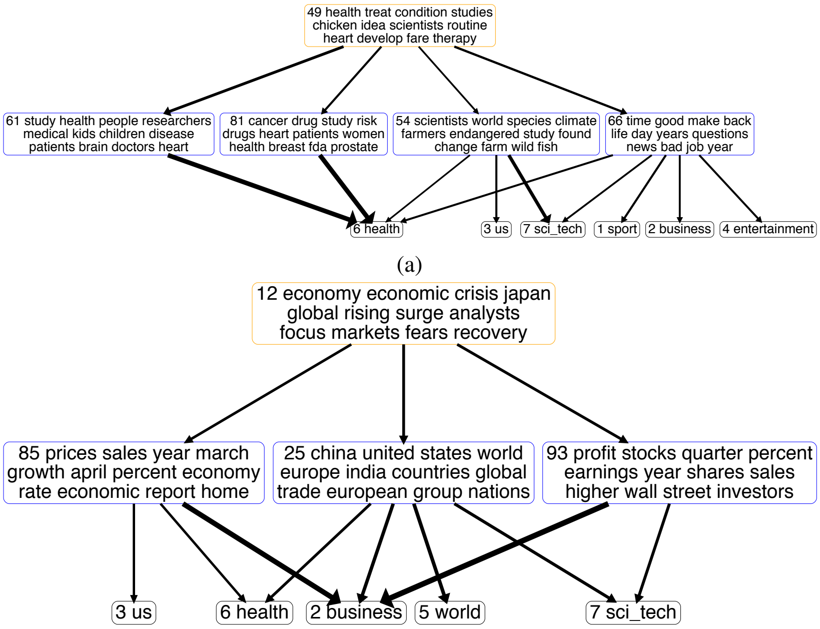
score from the top 10 words of each topic and reported scores averaged over top 50 topics with highest NPMI, where “rubbish” topics are eliminated, following Yang et al. (2015)[7]. In the training documents, we further varied the proportion of the words used in training to mimic the case of sparse texts. All the models ran five times with different random seeds and we reported the averaged value with standard deviations.

The results of perplexity and topic coherence are shown in Figure 3 and Table 1, respectively. We have the following remarks on the results: **(1)** In general, for the models with DirBN, the performance is significantly improved compared with the counterparts without DirBN, especially in terms of perplexity and topic coherence and with low proportion of the training words. **(2)** In terms of all the measures, DirBN-2/3 always has better results than DirBN-1. Whereas if we compare GBN with PFA, its perplexity is worse than PFA's, especially for sparse texts. This demonstrates that hierarchical structures on θ (i.e., GBN) may not perform as well as hierarchical structures on ϕ (i.e., DirBN) on sparse texts. **(3)** Although PFA+DirBN-1 and PFA+Mallet both impose a symmetric Dirichlet on ϕ, the former usually has better perplexity. **(4)** The dual deep model (GBN+DirBN-3) usually performs the best on topic coherence, which demonstrates the benefits of the deep structures.

Qualitative analysis on topic hierarchies [8] GBN+DirBN is a dual deep model that discovers two sets of hierarchies, one induced by GBN on $\boldsymbol{\theta}$ and the other induced by DirBN on $\boldsymbol{\phi}$. The topics in the

[7] We used the Palmetto package (`http://palmetto.aksw.org`) with a large Wikipedia dump.

[8] More visualisations of topic hierarchies are shown in the supplementary material.


49 health treat condition studies chicken idea scientists routine heart develop fare therapy
61 study health people researchers medical kids children disease patients brain doctors heart
81 cancer drug study risk drugs heart patients women health breast fda prostate
54 scientists world species climate farmers endangered study found change farm wild fish
66 time good make back life day years questions news bad job year
6 health
3 us
7 sci_tech
1 sport
2 business
4 entertainment
12 economy economic crisis japan global rising surge analysts focus markets fears recovery
85 prices sales year march growth april percent economy rate economic report home
25 china united states world europe india countries global trade european group nations
93 profit stocks quarter percent earnings year shares sales higher wall street investors
3 us
6 health
2 business
5 world
7 sci_tech

(a)

(b)

Figure 4: Topic hierarchies discovered by MetaLDA+DirBN. The topics in the yellow and blue rectangles are the second and first layer topics in DirBN and the correlated labels to the first-layer topics are shown at the bottom of each figure. Thicker arrows indicate stronger correlations.

first layer of DirBN connect the two sets of hierarchies. In Table 2, we show the first-layer topics and the correlated second-layer topics in the two hierarchies. It is interesting to see that the second-layer topics of DirBN are more abstract. For example, the second topic is about teams and player in NBA, while its correlated second-layer topics are more general words for sports. Moreover, DirBN is able to discover layer-wise semantically meaningful topic correlations with fewer overlapping top words. This is because GBN combines the words in the first-layer topics to form the second-layer topics, whereas DirBN decomposes the first-layer topics into the second-layer ones.

In MetaLDA+DirBN, the MetaLDA part is able to use document labels to construct TPs (Zhao et al., 2017a), by learning a correlation matrix between the labels and topics, while the DirBN part learns the topic hierarchy. The first-layer topics of DirBN link the correlation matrix and the topic hierarchy together. Figure 4 shows the sample linkages between topic hierarchies and labels on TMN, where the documents are labelled with 7 categories: 1 sport, 2 business, 3 us, 4 entertainment, 5 world, 6 health, 7 sci-tech. One can observe that there is a well correspondence between the topic hierarchies and the labels.

6 Conclusions

We have presented DirBN, a multi-layer process generating word distributions of topics. With real topics in each layer, DirBN is able to discover interpretable topic hierarchies. As a flexible module, DirBN can be adapted to other advanced topic models and improve the performance and interpretability, especially on sparse texts. We have demonstrated DirBN's advantages by equipping PFA, MetaLDA, and GBN, with DirBN. With the help of data augmentation, the inference of DirBN can be done by a layer-wise Gibbs sampling, as a full conjugate model.

Future directions include deriving alternative inference algorithms, such as variational inference (Hoffman et al., 2013), conditional density filtering (Guhaniyogi et al., 2018), and stochastic gradient-based approaches (Chen et al., 2014; Ding et al., 2014; Welling and Teh, 2011).

Acknowledgments

M. Zhou acknowledges the support of Award IIS-1812699 from the U.S. National Science Foundation.

References

D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *JMLR*, vol. 3, pp. 993–1022, 2003.

D. M. Blei, T. L. Griffiths, and M. I. Jordan, "The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies," *Journal of the ACM*, vol. 57, no. 2, p. 7, 2010.

J. Paisley, C. Wang, D. M. Blei, and M. I. Jordan, "Nested hierarchical Dirichlet processes," *TPAMI*, vol. 37, no. 2, pp. 256–270, 2015.

G. E. Hinton and R. R. Salakhutdinov, "Replicated softmax: An undirected topic model," in *NIPS*, 2009, pp. 1607–1614.

H. Larochelle and S. Lauly, "A neural autoregressive topic model," in *NIPS*, 2012, pp. 2708–2716.

N. Srivastava, R. Salakhutdinov, and G. Hinton, "Modeling documents with a deep Boltzmann machine," in *UAI*, 2013, pp. 616–624.

A. Srivastava and C. Sutton, "Autoencoding variational inference for topic models," 2017.

Y. Miao, E. Grefenstette, and P. Blunsom, "Discovering discrete latent topics with neural variational inference," in *ICML*, 2017, pp. 2410–2419.

H. Zhang, B. Chen, D. Guo, and M. Zhou, "WHAI: Weibull hybrid autoencoding inference for deep topic modeling," 2018.

G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

Z. Gan, C. Chen, R. Henao, D. Carlson, and L. Carin, "Scalable deep Poisson factor analysis for topic modeling," in *ICML*, 2015, pp. 1823–1832.

R. Ranganath, L. Tang, L. Charlin, and D. Blei, "Deep exponential families," in *AISTATS*, 2015, pp. 762–771.

R. Henao, Z. Gan, J. Lu, and L. Carin, "Deep Poisson factor modeling," in *NIPS*, 2015, pp. 2800–2808.

M. Zhou, Y. Cong, and B. Chen, "Augmentable gamma belief networks," *JMLR*, vol. 17, no. 163, pp. 1–44, 2016.

J. D. Mcauliffe and D. M. Blei, "Supervised topic models," in *NIPS*, 2008, pp. 121–128.

M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *UAI*, 2004, pp. 487–494.

M. Zhou, L. Hannah, D. B. Dunson, and L. Carin, "Beta-negative binomial process and Poisson factor analysis," in *AISTATS*, 2012, pp. 1462–1471.

H. Zhao, L. Du, W. Buntine, and G. Liu, "Metalda: A topic model that efficiently incorporates meta information," in *ICDM*, 2017, pp. 635–644.

J. D. Lafferty and D. M. Blei, "Correlated topic models," in *NIPS*, 2006, pp. 147–154.

Y. Teh, M. Jordan, M. Beal, and D. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2012.

Y. Tang and R. R. Salakhutdinov, "Learning stochastic feedforward neural networks," in *NIPS*, 2013, pp. 530–538.

M. Zhou, "Infinite edge partition models for overlapping community detection and link prediction," in *AISTATS*, 2015, pp. 1135—–1143.

M. Zhou and L. Carin, “Negative binomial process count and mixture modeling,” *TPAMI*, vol. 37, no. 2, pp. 307–320, 2015.

H. Zhao, L. Du, and W. Buntine, “Leveraging node attributes for incomplete relational data,” in *ICML*, 2017, pp. 4072–4081.

M. Zhou, “Nonparametric Bayesian negative binomial factor analysis,” *Bayesian Analysis*, 2018.

H. Zhao, L. Du, W. Buntine, and G. Liu, “Leveraging external information in topic modelling,” *KAIS*, pp. 1–33, 2018.

M. Zhou, Y. Cong, and B. Chen, “The Poisson gamma belief network,” in *NIPS*, 2015, pp. 3043–3051.

H. M. Wallach, D. M. Mimno, and A. McCallum, “Rethinking LDA: Why priors matter,” in *NIPS*, 2009, pp. 1973–1981.

H. Zhao, L. Du, and W. Buntine, “A word embeddings informed focused topic model,” in *ACML*, 2017, pp. 423–438.

H. Zhao, L. Du, W. Buntine, and M. Zhou, “Inter and intra topic structure learning with word embeddings,” in *ICML*, 2018, pp. 5887–5896.

I. Sato and H. Nakagawa, “Topic models with power-law using Pitman-Yor process,” in *SIGKDD*, 2010, pp. 673–682.

W. L. Buntine and S. Mishra, “Experiments with non-parametric topic models,” in *SIGKDD*, 2014, pp. 881–890.

C. Chen, W. Buntine, N. Ding, L. Xie, and L. Du, “Differential topic models,” *TPAMI*, vol. 37, no. 2, pp. 230–242, 2015.

R. V. Lindsey, W. P. Headden III, and M. J. Stipicevic, “A phrase-discovering topic model using hierarchical Pitman-Yor processes,” in *EMNLP*, 2012, pp. 214–222.

C. Archambeau, B. Lakshminarayanan, and G. Bouchard, “Latent IBP compound Dirichlet allocation,” *TPAMI*, vol. 37, no. 2, pp. 321–333, 2015.

F. Wood and Y. W. Teh, “A hierarchical nonparametric Bayesian approach to statistical language model domain adaptation,” in *AISTATS*, 2009, pp. 607–614.

L. Du, W. Buntine, and H. Jin, “Modelling sequential text with an adaptive topic model,” in *EMNLP*, 2012, pp. 535–545.

J. H. Kim, D. Kim, S. Kim, and A. Oh, “Modeling topic hierarchies with the recursive chinese restaurant process,” in *CIKM*, 2012, pp. 783–792.

A. Ahmed, L. Hong, and A. Smola, “Nested Chinese restaurant franchise process: Applications to user tracking and document modeling,” in *ICML*, 2013, pp. 1426–1434.

W. Li and A. McCallum, “Pachinko allocation: DAG-structured mixture models of topic correlations,” in *ICML*, 2006, pp. 577–584.

Y. Cong, B. Chen, H. Liu, and M. Zhou, “Deep latent Dirichlet allocation with topic-layer-adaptive stochastic gradient Riemannian MCMC,” in *ICML*, 2017, pp. 864–873.

N. Aletras and M. Stevenson, “Evaluating topic coherence using distributional semantics,” in *Proc. of the 10th Intnl. Conf. on Computational Semantics*, 2013, pp. 13–22.

J. H. Lau, D. Newman, and T. Baldwin, “Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality,” in *EACL*, 2014, pp. 530–539.

Y. Yang, D. Downey, and J. Boyd-Graber, “Efficient methods for incorporating knowledge into topic models,” in *EMNLP*, 2015, pp. 308–317.

M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, “Stochastic variational inference,” *JMLR*, vol. 14, no. 1, pp. 1303–1347, 2013.

R. Guhaniyogi, S. Qamar, and D. B. Dunson, “Bayesian conditional density filtering,” *Journal of Computational and Graphical Statistics*, 2018.

T. Chen, E. Fox, and C. Guestrin, “Stochastic gradient Hamiltonian Monte Carlo,” in *ICML*, 2014, pp. 1683–1691.

N. Ding, Y. Fang, R. Babbush, C. Chen, R. D. Skeel, and H. Neven, “Bayesian sampling using stochastic gradient thermostats,” in *NIPS*, 2014, pp. 3203–3211.

M. Welling and Y. W. Teh, “Bayesian learning via stochastic gradient Langevin dynamics,” in *ICML*, 2011, pp. 681–688.

Supplementary materials for “Dirichlet belief networks for topic structure learning”

He Zhao[1], Lan Du[1], Wray Buntine[1], and Mingyuan Zhou[2]
[1]Faculty of Information Technology, Monash University, Australia
[2]McCombs School of Business, The University of Texas at Austin, USA

1 Details of the inference

Given the latent counts $\boldsymbol{x}^{(t)}_{k_t}$, the details of inference of the the t-th ($t < T$) layer of DirBN are as follows:

$$\boldsymbol{\phi}^{(t)}_{k_t} \sim \text{Dir}\left(\boldsymbol{\psi}^{(t)}_{k_t} + \boldsymbol{x}^{(t)}_{k_t}\right), \tag{1}$$

$$q^{(1)}_{k_1} \sim \text{Beta}\left(\psi^{(1)}_{\cdot k_1}, x^{(1)}_{\cdot k_1}\right), \tag{2}$$

$$y^{(t)}_{vk_t} \sim \text{CRT}\left(x^{(t)}_{vk_t}, \psi^{(t)}_{vk_t}\right), \tag{3}$$

$$\left(z^{(t)}_{v1k_t}, \cdots, z^{(t)}_{vK_{t+1}k_t}\right) \sim \text{Mult}\left(y^{(t)}_{vk_t}, \frac{\phi^{(t+1)}_{v1}\beta^{(t)}_{1k_t}}{\psi^{(t)}_{vk_t}}, \cdots, \frac{\phi^{(t+1)}_{vK_{t+1}}\beta^{(t)}_{K_{t+1}k_t}}{\psi^{(t)}_{vk_t}}\right), \tag{4}$$

$$\beta^{(t)}_{k_{t+1}k_t} \sim \text{Ga}\left(\gamma^{(t)}_{k_{t+1}} + z^{(t)}_{\cdot k_{t+1}k_t}, 1.0\right) / \left(c^{(t)} - \log q^{(t)}_{k_t}\right), \tag{5}$$

$$m^{(t)}_{k_{t+1}k_t} \sim \text{CRT}\left(z^{(t)}_{\cdot k_{t+1}k_t}, \gamma^{(t)}_{k_{t+1}}\right), \tag{6}$$

$$\gamma^{(t)}_{k_{t+1}} \sim \text{Ga}\left(\gamma^{(t)}_0/K_{t+1} + \sum_{k_t}^{K_t} m^{(t)}_{k_{t+1}k_t}, 1.0\right) / \left(c^{(t)}_0 + n^{(t)}\right), \tag{7}$$

$$c^{(t)} \sim \text{Ga}\left(g_0 + K_t\sum_{k_{t+1}}^{K_{t+1}} \gamma^{(t)}_{k_{t+1}}, 1.0\right) / \left(h_0 + \sum_{k_{t+1},k_t}^{K_{t+1},K_t} \beta^{(t)}_{k_{t+1}k_t}\right), \tag{8}$$

$$p^{(t)}_{k_{t+1}} \sim \text{CRT}\left(\sum_{k_t}^{K_t} m^{(t)}_{k_{t+1}k_t}, \gamma^{(t)}_0/K_{t+1}\right), \tag{9}$$

$$\gamma^{(t)}_0 \sim \text{Ga}\left(e_0 + \sum_{k_{t+1}}^{K_{t+1}} p^{(t)}_{k_{t+1}}, 1.0\right) / \left(f_0 + \log\frac{n^{(t)} + c^{(t)}_0}{c^{(t)}_0}\right), \tag{10}$$

$$c^{(t)}_0 \sim \text{Ga}\left(g_0 + \gamma^{(t)}_0, 1.0\right) / \left(h_0 + \sum_{k_{t+1}} \gamma^{(t)}_{k_{t+1}}\right), \tag{11}$$

where $n^{(t)} = \sum_{k_t}^{K_t} \log\frac{c^{(t)} - \log q^{(t)}_{k_t}}{c^{(t)}}$.
In the top layer $t = T$, we have:

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

$$\boldsymbol{\phi}_{k_T}^{(T)} \sim \text{Dir}\left(\eta + \boldsymbol{x}_{k_T}^{(T)}\right), \tag{12}$$

$$s_{vk_T} \sim \text{CRT}\left(x_{vk_T}^{(T)}, \eta\right), \tag{13}$$

$$\eta \sim \text{Ga}\left(a_0 + \sum_{v,k_T}^{V,K_T} s_{vK_T}, 1.0\right) / \left(b_0 - \sum_{k_T}^{K_T} \log q_{k_T}^{(T)}\right). \tag{14}$$

The inference process of DirBN is in Algorithm 1. Note that in different models, after the topic assignments of words are obtained, the inference of DirBN is the same.

2 Details of the combined models

PFA+DirBN The generative process of PFA+DirBN is shown as follows:

$$p_k \sim \text{Beta}\left(c\epsilon, c(1-\epsilon)\right), r_k \sim \text{Ga}(c_0 r_0, 1/c_0), \theta_{kd} \sim \text{Ga}\left(r_k, \frac{p_k}{1-p_k}\right),$$

$$\boldsymbol{\phi}_k \sim \text{DirBN}(T), x_{vd} = \sum_k^K x_{vdk}, x_{vdk} \sim \text{Pois}(\phi_{vk}\theta_{kd}), \tag{15}$$

where DirBN(T) stands for the generative process of DirBN with T layers.

MetaLDA+DirBN The generative process of MetaLDA+DirBN is as follows:

$$\lambda_{lk} \sim \text{Ga}(a_0, 1/b_0), \alpha_{kd} = \prod_l^L (\lambda_{lk})^{f_{ld}}, \boldsymbol{\theta}_d \sim \text{Dir}(\boldsymbol{\alpha}_d), \boldsymbol{\phi}_k \sim \text{DirBN}(T),$$

$$z_{id} \sim \text{Categorical}(\boldsymbol{\theta}_d), w_{id} \sim \text{Categorical}(\boldsymbol{\phi}_{z_{id}}), \tag{16}$$

where L is the number of unique document labels, $l \in \{1, \cdots, L\}$, $f_{ld} \in \{0, 1\}$ indicates whether document d has label l, $w_{id} = v$ is the i-th word in document d, and $z_{id} = k$ is the topic assignment of w_{id}.

GBN+DirBN The generative process of GBN+DirBN is as follows:

$$\boldsymbol{\theta}_d^{(S)} \sim \text{Ga}\left(\boldsymbol{r}, 1/c_j^{(T+1)}\right), \cdots, \boldsymbol{\theta}_d^{(s)} \sim \text{Ga}\left(\widetilde{\boldsymbol{\Phi}}^{(s+1)}\boldsymbol{\theta}_d^{(s+1)}, 1/c_d^{(s+1)}\right),$$

$$\cdots$$

$$\boldsymbol{\theta}_d^{(1)} \sim \text{Ga}\left(\widetilde{\boldsymbol{\Phi}}^{(2)}\boldsymbol{\theta}_d^{(2)}, p_d^{(2)}/(1-p_d^{(2)})\right), \boldsymbol{\phi}_k \sim \text{DirBN}(T),$$

$$x_{vd} = \sum_k^K x_{vdk}, x_{vdk} \sim \text{Pois}(\phi_{vk}\theta_{kd}^{(1)}), \tag{17}$$

where $s \in \{1, \cdots, S\}$ is the index of the s layer in GBN.

3 More results

For document classification, the TPs were used as input features for a L_2 regularized logistic regression using the LIBLINEAR package to predict the document labels. We used the same train/test splits as in perplexity evaluation, except that all the words in a test document were used to infer its TP. The results on WS and TMN are shown in Table 1.

4 Visualisation of topic hierarchies

Shown in Figure 2 to 5.

Table 1: Document classification

	WS			TMN		
Training words	20%	40%	100%	20%	40%	100%
PFA	67.58±5.73	81.08±0.83	82.29±0.73	73.02±1.43	78.68±0.28	80.00±0.51
PFA+mallet	73.97±1.12	79.64±0.89	82.75±0.89	72.84±0.40	78.02±0.89	79.92±0.66
PFA+DirBN-1	77.11±0.55	81.69±0.53	82.26±0.48	73.08±0.33	78.40±0.31	79.77±0.56
PFA+DirBN-3	76.74±0.57	82.04±0.28	83.68±1.04	74.41±0.60	78.99±0.46	79.91±0.56
MetaLDA	67.94±3.00	**83.26**±1.21	84.18±1.10	74.02±0.62	78.88±0.27	80.04±0.49
MetaLDA + DirBN-1	76.67±0.88	81.38±1.02	83.07±0.70	74.10±0.22	79.67±0.67	80.63±0.10
MetaLDA + DirBN-3	77.84±1.06	82.53±0.46	83.97±1.09	75.03±0.26	79.37±0.63	80.99±0.22
GBN	68.87±4.67	82.97±0.49	84.35±0.91	72.88±1.08	79.28±0.41	**81.44**±0.21
GBN+DirBN-1	76.73±0.70	82.54±0.81	83.18±0.40	74.42±0.32	79.59±0.30	80.87±0.68
GBN+DirBN-3	**78.17**±1.88	82.82±1.08	**84.28**±1.12	**75.36**±0.60	**79.79**±0.48	81.10±0.34

```
Require: $\boldsymbol{x}_{k_1}^{(1)}$ for all $k_1, T(T > 1), a_0, b_0, e_0, f_0, g_0, h_0$ $MaxIteration$
Ensure: $\boldsymbol{\beta}_{k_t}^{(t)}, \boldsymbol{\phi}_{k_t}^{(t)}$ for all $k_t$
Randomly initialise all the latent variables according to the generative process
for $iter \leftarrow 1$ to $MaxIteration$ do
    $/ *$ Propagating the latent counts from the bottom up $* /$
    for $t \leftarrow 1$ to $T$ do
        for all $k_t$ and $v$ do
            Sample $y_{vk_t}^{(t)}$ by Eq. (3)
            for all $k_{t+1}$ do
                Sample $z_{vk_{t+1}k_t}^{(t)}$ by Eq. (4)
            end for
        end for
    end for
    $/ *$ Updating the latent variables from the top down $* /$
    for $t \leftarrow T$ to $1$ do
        if $t = T$ then
            for all $k_T$ and $v$ do
                Sample $s_{vk_T}$ by Eq. (13)
            end for
            Sample $\eta$ by Eq. (14)
            for all $k_T$ do
                Sample $\boldsymbol{\phi}_{k_T}^{(T)}$ by Eq. (12)
            end for
        else
            for all $k_t$ do
                Compute $\boldsymbol{\psi}_{k_t}^{(t)}$ by $\boldsymbol{\psi}_{k_t}^{(t)} = \sum_{k_{t+1}}^{K_{t+1}} \boldsymbol{\phi}_{k_{t+1}}^{(t+1)} \beta_{k_{t+1}k_t}^{(t)}$
            end for
            for all $k_t$ do
                Sample $q_{k_t}^{(t)}$ by Eq. (2)
            end for
            for all $k_t$ and $k_{t+1}$ do
                Sample $m_{k_{t+1}k_t}^{(t)}$ by Eq. (6)
            end for
            for all $k_{t+1}$ do
                Sample $\gamma_{k_{t+1}}^{(t)}, p_{k_{t+1}}^{(t)}$ by Eq. (7,9)
            end for
            Sample $c^{(t)}, \gamma_0^{(t)}, c_0^{(t)}$ by Eq. (8,10,11)
            for all $k_t$ and $k_{t+1}$ do
                Sample $\beta_{k_{t+1}k_t}^{(t)}$ by Eq. (5)
            end for
            for all $k_t$ do
                Sample $\boldsymbol{\phi}_{k_t}^{(t)}$ by Eq. (1)
            end for
        end if
    end for
end for
```

Figure 1: Infernece algorithm for DirBN

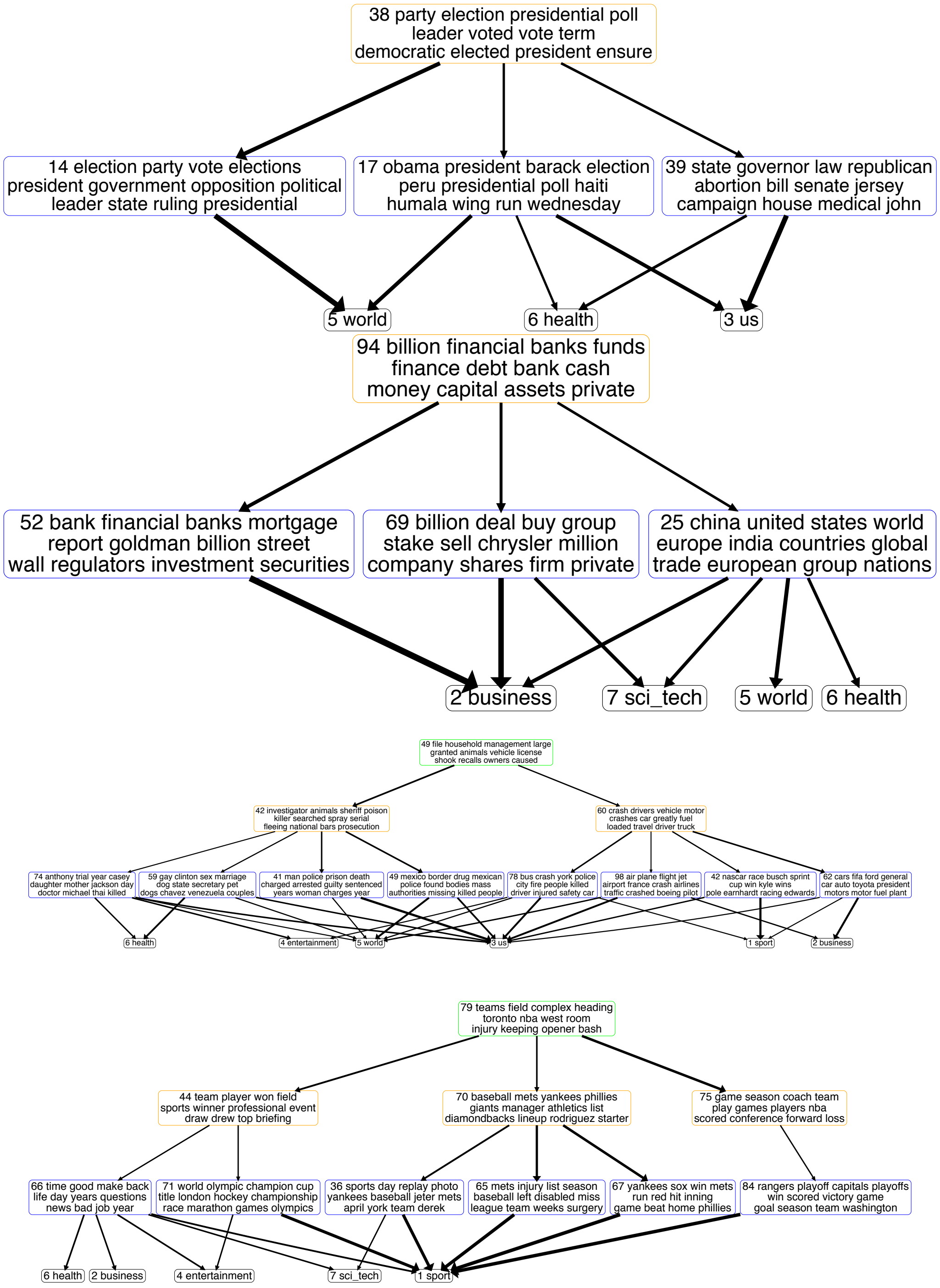

38 party election presidential poll leader voted vote term democratic elected president ensure
14 election party vote elections president government opposition political leader state ruling presidential
17 obama president barack election peru presidential poll haiti humala wing run wednesday
39 state governor law republican abortion bill senate jersey campaign house medical john
5 world
6 health
3 us
94 billion financial banks funds finance debt bank cash money capital assets private
52 bank financial banks mortgage report goldman billion street wall regulators investment securities
69 billion deal buy group stake sell chrysler million company shares firm private
25 china united states world europe india countries global trade european group nations
2 business
7 sci_tech
5 world
6 health
49 file household management large granted animals vehicle license shook recalls owners caused
42 investigator animals sheriff poison killer searched spray serial fleeing national bars prosecution
60 crash drivers vehicle motor crashes car greatly fuel loaded travel driver truck
74 anthony trial year casey daughter mother jackson day doctor michael thai killed
59 gay clinton sex marriage dog state secretary pet dogs chavez venezuela couples
41 man police prison death charged arrested guilty sentenced years woman charges year
49 mexico border drug mexican police found bodies mass authorities missing killed people
78 bus crash york police city fire people killed driver injured safety car
98 air plane flight jet airport france crash airlines traffic crashed boeing pilot
42 nascar race busch sprint cup win kyle wins pole earnhardt racing edwards
62 cars fifa ford general car auto toyota president motors motor fuel plant
6 health
4 entertainment
5 world
3 us
1 sport
2 business
79 teams field complex heading toronto nba west room injury keeping opener bash
44 team player won field sports winner professional event draw drew top briefing
70 baseball mets yankees phillies giants manager athletics list diamondbacks lineup rodriguez starter
75 game season coach team play games players nba scored conference forward loss
66 time good make back life day years questions news bad job year
71 world olympic champion cup title london hockey championship race marathon games olympics
36 sports day replay photo yankees baseball jeter mets april york team derek
65 mets injury list season baseball left disabled miss league team weeks surgery
67 yankees sox win mets run red hit inning game beat home phillies
84 rangers playoff capitals playoffs win scored victory game goal season team washington
6 health
2 business
4 entertainment
7 sci_tech
1 sport

Figure 2: Topic hierarchies discovered by MetaLDA+DirBN on TMN. The topics in the green, yellow, and blue rectangles are the third, second, and first layer topics in DirBN and the correlated document labels are shown on the bottom of each figure. Thicker arrows indicate stronger correlations.

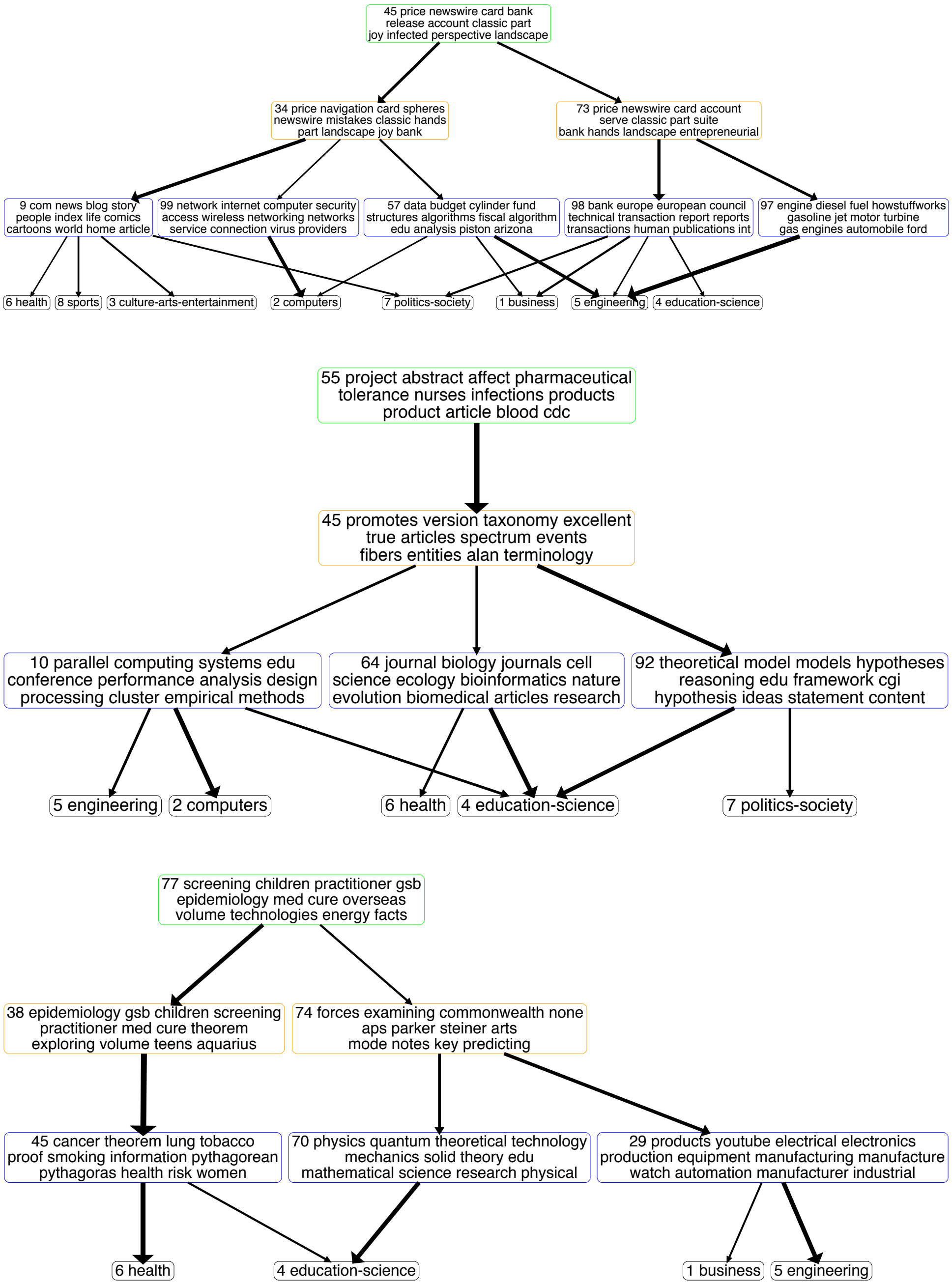

45 price newswire card bank release account classic part joy infected perspective landscape
34 price navigation card spheres newswire mistakes classic hands part landscape joy bank
73 price newswire card account serve classic part suite bank hands landscape entrepreneurial
9 com news blog story people index life comics cartoons world home article
99 network internet computer security access wireless networking networks service connection virus providers
57 data budget cylinder fund structures algorithms fiscal algorithm edu analysis piston arizona
98 bank europe european council technical transaction report reports transactions human publications int
97 engine diesel fuel howstuffworks gasoline jet motor turbine gas engines automobile ford
6 health
8 sports
3 culture-arts-entertainment
2 computers
7 politics-society
1 business
5 engineering
4 education-science
55 project abstract affect pharmaceutical tolerance nurses infections products product article blood cdc
45 promotes version taxonomy excellent true articles spectrum events fibers entities alan terminology
10 parallel computing systems edu conference performance analysis design processing cluster empirical methods
64 journal biology journals cell science ecology bioinformatics nature evolution biomedical articles research
92 theoretical model models hypotheses reasoning edu framework cgi hypothesis ideas statement content
5 engineering
2 computers
6 health
4 education-science
7 politics-society
77 screening children practitioner gsb epidemiology med cure overseas volume technologies energy facts
38 epidemiology gsb children screening practitioner med cure theorem exploring volume teens aquarius
74 forces examining commonwealth none aps parker steiner arts mode notes key predicting
45 cancer theorem lung tobacco proof smoking information pythagorean pythagoras health risk women
70 physics quantum theoretical technology mechanics solid theory edu mathematical science research physical
29 products youtube electrical electronics production equipment manufacturing manufacture watch automation manufacturer industrial
6 health
4 education-science
1 business
5 engineering

Figure 3: Topic hierarchies discovered by MetaLDA+DirBN on WS. The topics in the green, yellow, and blue rectangles are the third, second, and first layer topics in DirBN and the correlated document labels are shown on the bottom of each figure. Thicker arrows indicate stronger correlations.

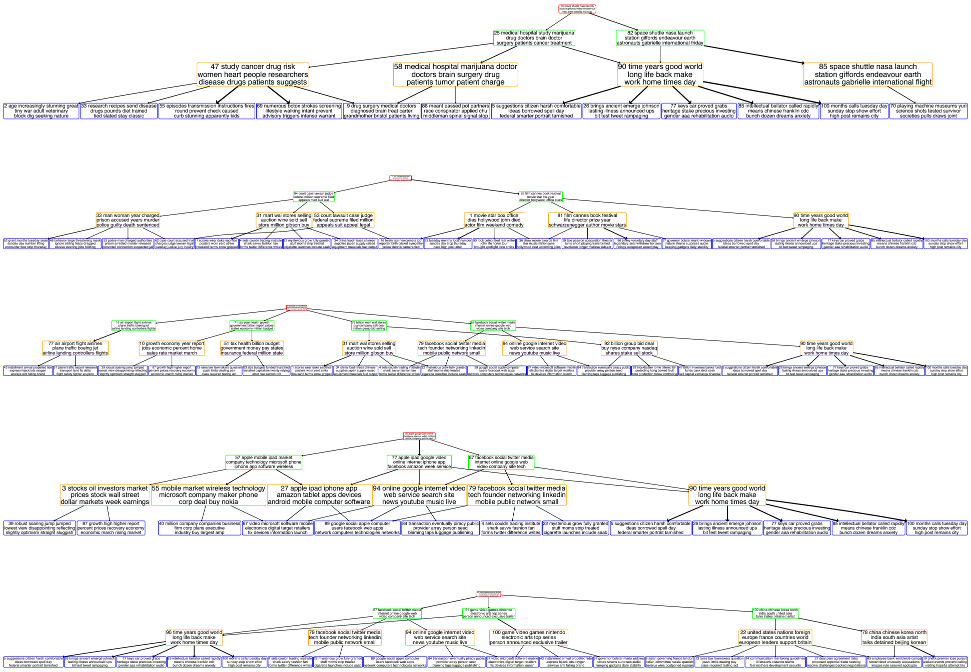

25 medical hospital study marijuana drug doctors brain doctor surgery patients cancer treatment
82 space shuttle nasa launch station giffords endeavour earth astronauts gabrielle international friday
47 study cancer drug risk women heart people researchers disease drugs patients suggests
58 medical hospital marijuana doctor doctors brain surgery drug patients tumor patient charge
90 time years good world long life back make work home times day
85 space shuttle nasa launch station giffords endeavour earth astronauts gabrielle international flight
2 age increasingly stunning great tiny war adult veterinary block dig seeking nature
33 research recipes send disease drugs pounds diet trained tied slated stay classic
55 episodes transmission instructions fires round prevent check caused curb stunning apparently kids
69 numerous botox strokes screening lifestyle walking infant prevent advisory triggers intense warrant
9 drug surgery medical doctors diagnosed brain treat carter grandmother bristol patients living
88 meant passed pot partners race conspirator applied chu middleman spinal signal stop
5 suggestions citizen harsh comfortable ideas borrowed spell day federal smarter portrait tarnished
26 brings ancient emerge johnson lasting illness announced ups bit test tweet rampaging
77 keys car proved grabs heritage stake precious investing gender aaa rehabilitation audio
85 intellectual bellator called rapidly means chinese franklin cdc bunch dozen dreams anxiety
100 months calls tuesday day sunday stop show effort high post remains city
70 playing machine museums yuri science shots tested survivor societies pulls draws joint
94 court case lawsuit judge federal million supreme filed appeals mart suit wal
62 film cannes book festival movie star life year director hollywood office stars
33 man woman year charged prison accused years murder police guilty death sentenced
31 mart wal stores selling auction wine sold sell store million gibson buy
53 court lawsuit case judge federal supreme filed million appeals suit appeal legal
1 movie star box office dies hollywood john died actor film weekend comedy
81 film cannes book festival life director prize year schwarzenegger author movie stars
90 time years good world long life back make work home times day
16 air airport flight airlines plane traffic boeing jet airline landing controllers flights
71 tax year health growth government billion report prices states economy million budget
79 billion mart wal stores buy company sell deal million group bid selling
87 facebook social twitter media internet online google web video company site tech
77 air airport flight airlines plane traffic boeing jet airline landing controllers flights
10 growth economy year report jobs economic percent home sales rate market march
51 tax health billion budget government money pay states insurance federal million state
31 mart wal stores selling auction wine sold sell store million gibson buy
79 facebook social twitter media tech founder networking linkedin mobile public network small
94 online google internet video web service search site news youtube music live
92 billion group bid deal buy nyse company nasdaq shares stake sell stock
90 time years good world long life back make work home times day
57 apple mobile ipad market company technology microsoft phone iphone app software wireless
77 apple ipad google video online internet iphone app facebook amazon week service
87 facebook social twitter media internet online google web video company site tech
3 stocks oil investors market prices stock wall street dollar markets week earnings
55 mobile market wireless technology microsoft company maker phone corp deal buy nokia
27 apple ipad iphone app amazon tablet apps devices android mobile computer software
94 online google internet video web service search site news youtube music live
79 facebook social twitter media tech founder networking linkedin mobile public network small
90 time years good world long life back make work home times day
39 robust soaring jump jumped lowest view disappointing reflecting slightly optimism straight sluggish
87 growth high higher report percent prices recovery economy economic march rising market
40 million company companies business firm corp plans executive industry buy largest amp
67 video microsoft software mobile electronics digital target retailers fix devices information launch
89 google social apple computer users facebook web apps network computers technologies networks
84 transaction eventually piracy public provider array person seed blaming taps luggage publishing
4 sets couldn trading institute shark savvy fashion fan forms twitter difference writes
22 mysterious grow fully granted stuff moms strip treated cigarette launches include saab
5 suggestions citizen harsh comfortable ideas borrowed spell day federal smarter portrait tarnished
26 brings ancient emerge johnson lasting illness announced ups bit test tweet rampaging
77 keys car proved grabs heritage stake precious investing gender aaa rehabilitation audio
85 intellectual bellator called rapidly means chinese franklin cdc bunch dozen dreams anxiety
100 months calls tuesday day sunday stop show effort high post remains city
87 facebook social twitter media internet online google web video company site tech
51 game video games nintendo electronic arts top series person announced exclusive trailer
100 china chinese korea north india south united asia talks states detained artist
90 time years good world long life back make work home times day
79 facebook social twitter media tech founder networking linkedin mobile public network small
94 online google internet video web service search site news youtube music live
100 game video games nintendo electronic arts top series person announced exclusive trailer
22 united states nations foreign europe france countries world european leaders support britain
78 china chinese korea north india south asia artist talks detained beijing korean

Figure 4: Topic hierarchies discovered by GBN+DirBN on TMN. The topics in the red and green rectangles are the third and second-layer topics discovered by GBN on TPs. The topics in the blue rectangles are the second-layer topics discovered by DirBN on WDs. The topics in the yellow rectangles are the first-layer topics connecting the higher-layer topics of GBN and DirBN.

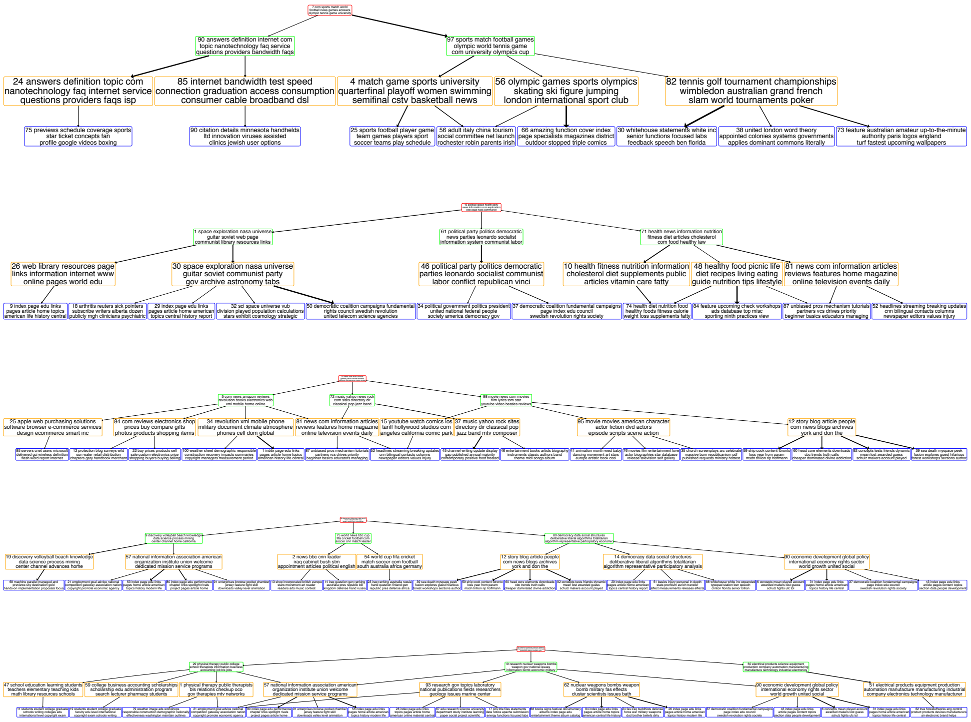

7 com sports match world football news games answers olympic tennis game university
90 answers definition internet com topic nanotechnology faq service questions providers bandwidth faqs
97 sports match football games olympic world tennis game com university olympics cup
24 answers definition topic com nanotechnology faq internet service questions providers faqs isp
85 internet bandwidth test speed connection graduation access consumption consumer cable broadband dsl
4 match game sports university quarterfinal playoff women swimming semifinal cstv basketball news
56 olympic games sports olympics skating ski figure jumping london international sport club
82 tennis golf tournament championships wimbledon australian grand french slam world tournaments poker
75 previews schedule coverage sports star ticket concepts fan profile google videos boxing
90 citation details minnesota handhelds ltd innovation viruses assisted clinics jewish user options
25 sports football player game team games players sport soccer teams play schedule
56 adult italy china tourism social committee net launch rochester robin parents irish
66 amazing function cover index page specialists magazines district outdoor stopped triple comics
30 whitehouse statements white inc senior functions focused labs feedback speech ben florida
38 united london word theory appointed colonies systems governments applies dominant commons literally
73 feature australian amateur up-to-the-minute authority paris logos england turf fastest upcoming wallpapers
15 political space health party news information com exploration web page nasa communist
1 space exploration nasa universe guitar soviet web page communist library resources links
61 political party politics democratic news parties leonardo socialist information system communist labor
71 health news information nutrition fitness diet articles cholesterol com food healthy law
26 web library resources page links information internet www online pages world edu
30 space exploration nasa universe guitar soviet communist party gov archive astronomy tabs
46 political party politics democratic parties leonardo socialist communist labor conflict republican vinci
10 health fitness nutrition information cholesterol diet supplements public articles vitamin care fatty
48 healthy food picnic life diet recipes living eating guide nutrition tips lifestyle
81 news com information articles reviews features home magazine online television events daily
9 index page edu links pages article home topics american life history central
18 arthritis reuters sick pointers subscribe writers alberta dozen publicly mgh clinicians psychiatric
29 index page edu links pages article home american topics central history report
32 sci space universe vub division played population calculations stars exhibit cosmology strategic
50 democratic coalition campaigns fundamental rights council swedish revolution united telecom science agencies
34 political government politics president united national federal people society america democracy gov
37 democratic coalition fundamental campaigns page index edu council swedish revolution rights society
74 health diet nutrition food healthy foods fitness calorie weight loss supplements fatty
84 feature upcoming check workshops ads database top misc sporting ninth practices view
87 unbiased pros mechanism tutorials partners vcs drives priority beginner basics educators managing
52 headlines streaming breaking updates cnn bilingual contacts columns newspaper editors values injury
5 com news amazon reviews revolution books electronics web xml mobile home online
72 music yahoo news rock com sites directory dir classical pop jazz band
98 movie news com movies film lyrics tom star youtube video beatles reviews
25 apple web purchasing solutions software browser e-commerce services design ecommerce smart inc
84 com reviews electronics shop prices buy compare gifts photos products shopping items
34 revolution xml mobile phone military document climate atmosphere phones cell dom global
81 news com information articles reviews features home magazine online television events daily
15 youtube watch comics los tariff hollywood studios com angeles california comic park
37 music yahoo rock sites directory dir classical pop jazz band mtv composer
95 movie movies american character actor fiction dvd actors episode scripts scene action
12 story blog article people com news blogs archives york and don the
19 discovery volleyball beach knowledge data science process mining center channel advances home
57 national information association american organization institute union welcome dedicated mission service programs
2 news bbc cnn leader iraq cabinet bush stm appointment articles political english
54 world cup fifa cricket match soccer com football south australia africa germany
12 story blog article people com news blogs archives york and don the
14 democracy data social structures deliberative liberal algorithms totalitarian algorithm representative participatory analysis
80 economic development global policy international economy rights sector world growth united social
47 school education learning students teachers elementary teaching kids math library resources schools
59 college business accounting scholarships scholarship edu administration program search lecturer pharmacy students
1 physical therapy public therapists site relations checkup occ gov therapies mhi networks
57 national information association american organization institute union welcome dedicated mission service programs
93 research gov topics laboratory national publications fields researchers geology issues marine center
62 nuclear weapons bombs weapon bomb military fas effects cluster scientists issues biatb
80 economic development global policy international economy rights sector world growth united social
51 electrical products equipment production automation manufacture manufacturing industrial company electronics technology manufacturer

Figure 5: Topic hierarchies discovered by GBN+DirBN on WS. The topics in the red and green rectangles are the third and second-layer topics discovered by GBN on TPs. The topics in the blue rectangles are the second-layer topics discovered by DirBN on WDs. The topics in the yellow rectangles are the first-layer topics connecting the higher-layer topics of GBN and DirBN.

Inter and Intra Topic Structure Learning with Word Embeddings

He Zhao [1] Lan Du [1] Wray Buntine [1] Mingyaun Zhou [2]

Abstract

One important task of topic modeling for text analysis is interpretability. By discovering structured topics one is able to yield improved interpretability as well as modeling accuracy. In this paper, we propose a novel topic model with a deep structure that explores both inter-topic and intra-topic structures informed by word embeddings. Specifically, our model discovers inter topic structures in the form of topic hierarchies and discovers intra topic structures in the form of sub-topics, each of which is informed by word embeddings and captures a fine-grained thematic aspect of a normal topic. Extensive experiments demonstrate that our model achieves the state-of-the-art performance in terms of perplexity, document classification, and topic quality. Moreover, with topic hierarchies and sub-topics, the topics discovered in our model are more interpretable, providing an illuminating means to understand text data.

1. Introduction

Significant research effort has been devoted to developing advanced text analysis technologies. Probabilistic topic models such as Latent Dirichlet Allocation (LDA), are popular approaches for this task, which discover latent topics from text collections. One preferred property of probabilistic topic models is interpretability: one can explain that a document is composed of topics and a topic is described by words. Although widely used, most variations of standard vanilla topic models (e.g., LDA) assume topics are independent and there are no structures among them. This limits those models' ability to explore any hierarchical thematic structures. Therefore, it is interesting to develop a model that is capable of exploring topic structures and yields not only improved modeling accuracy but also better interpretability.

Table 1. Example local topics with top 10 words

1	journal science biology research journals international cell psychology scientific bioinformatics
2	fitness piano guitar swimming violin weightlifting lessons training swim weight
3	taylor prince swift william jovi bon woman gala pill jon
4	san auto theft grand andreas mobile gta game rockstar december

One popular direction to explore topic structure is using the hierarchical/deep representation of text data, such as the nested hierarchical Dirichlet process (nHDP) (Paisley et al., 2015), Deep Poisson Factor Analysis (DPFA) (Gan et al., 2015), and Gamma Belief Network (GBN) (Zhou et al., 2016; Cong et al., 2017). In general, these models assume that topics in the higher layers of a hierarchy are more general/abstract than those in the lower layers. Therefore, by revealing hierarchical correlations between topics, topic hierarchies provide an intuitive way to understand text data.

In addition to topic hierarchies, we are also interested in analyzing the fine-grained thematic structure within each individual topic. As we know, in conventional models, topics are discovered locally from the word co-occurrences in a corpus. So we refer those topics as *local topics*. Due to the limitation of the context of a target corpus, some local topics may be hard to interpret because of the following two effects: (1) They can mix the words which co-occur locally in the target corpus but are less semantically related in general; (2) Local topics can be dominated by specialized words, which are less interpretable without extra knowledge. For example, we show four example topics of our experiments in Table 1, where we can see: Topic 1 is composed of the words from both the "scientific publication" and "biology" aspects; Topic 2 is a mixture of "sports" and "music"; Topics 3 and 4 are very specific topics about "singer" and "video game" respectively. We humans are able to understand those local topics in the above way because we are equipped with the global semantics of the words, making us go beyond the local context of the target corpus. Therefore, we are motivated to propose a model which is able to automatically analyze the fine-grained thematic structures of local topics, further improving the interpretability of topic modeling.

[1] Faculty of Information Technology, Monash University, Australia [2] McCombs School of Business, University of Texas at Austin. Correspondence to: Lan Du <lan.du@monash.edu>, Mingyuan Zhou <mingyuan.zhou@mccombs.utexas.edu>.

Proceedings of the 35^{th} *International Conference on Machine Learning*, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

Fortunately, word embeddings such as GloVe (Pennington et al., 2014), word2vec (Mikolov et al., 2013), and FastText (Bojanowski et al., 2017) can be used as an accessible source of global semantic information for topic models. Learned from large corpora, word embeddings encode the semantics of words with their locations in a space, where more related words are closer to each other. For example in Topic 1, according to the distances of word embeddings, the words "biology, cell, psychology, bioinformatics" should be in one cluster and "journal, science, research, international, scientific" should be in the other. Therefore, if a topic model can leverage the information in word embeddings, it may discover the fine-grained thematic structures of local topics. Furthermore, it has been demonstrated that conventional topic models suffer from data sparsity, resulting in a large performance degradation on some shorter internet-generated documents like tweets, product reviews, and news headlines (Zuo et al., 2016; Zhao et al., 2017c). In this case, word embeddings can also serve as complementary information to alleviate the sparsity issue in topic models.

In this paper, we propose a novel deep structured topic model, named the **W**ord **E**mbeddings **D**eep **T**opic **M**odel, WEDTM[1], which improves the interpretability of topic models by discovering topic hierarchies (i.e., *inter topic structure*) and fine-grained interpretations of local topics (i.e., *intra topic structure*). Specifically, the proposed model adapts a multi-layer Gamma Belief Network which generates deep representations of topics as well as documents. Moreover, WEDTM is able to split a local topic into a set of *sub-topics*, each of which captures one fine-grained thematic aspect of the local topic, in a way that each sub-topic is informed by word embeddings. WEDTM has the following key properties: **(1)** Better interpretability with topic hierarchies and sub-topics informed by word embeddings. **(2)** The state-of-the-art perplexity, document classification, and topic coherence performance, especially for sparse text data. **(3)** A straightforward Gibbs sampling algorithm facilitated by fully local conjugacy under data augmentation.

2. Related Work

Deep/hierarchical topic models: Several approaches have been developed to learn hierarchical representations of documents and topics. The Pachinko Allocation model (PAM) (Li & McCallum, 2006) assumes the topic structure is modeled by a directed acyclic graph (DAG), which is document specific. nCRP (Blei et al., 2010) models topic hierarchies by introducing a tree structure prior constructed with multiple CRPs. Paisley et al. (2015); Kim et al. (2012); Ahmed et al. (2013) further extend nCRP by either softening its constraints or applying it to different problems respectively. Poisson Factor Analysis (PFA) (Zhou et al., 2012) is a nonnegative matrix factorization model with Poisson link, which is a popular alternative to LDA, for topic modeling. The details of the close relationships between PFA and LDA can be found in Zhou (2018). There are several deep extensions to PFA for documents, such as DPFA (Gan et al., 2015), DPFM (Henao et al., 2015), and GBN (Zhou et al., 2016). Among them, GBN factorizes the factor score matrix (topic weights of documents) in PFA with nonnegative gamma-distributed hidden units connected by the weights drawn from the Dirichlet distribution. From a modeling perspective, GBN is related to PAM, while GBN assumes there is a corpus-level topic hierarchy shared by all the documents. As reported by Cong et al. (2017), GBN outperforms other hierarchical models including nHDP, DPFA, and DPFM. Despite having the attractive properties, these deep models barely consider intra topic structures or the sparsity issue associated with internet-generated corpora.

Word embedding topic models: Recently, there is a growing interest in applying word embeddings to topic models, especially for sparse data. For example, WF-LDA (Petterson et al., 2010) extends LDA to model word features with the logistic-normal transform, where word embeddings are used as word features in Zhao et al. (2017b). LF-LDA (Nguyen et al., 2015) integrates word embeddings into LDA by replacing the topic-word Dirichlet multinomial component with a mixture of a Dirichlet multinomial component and a word embedding component. Due to the non-conjugacy in WF-LDA and LF-LDA, part of the inference has to be done by MAP optimization. Instead of generating tokens, Gaussian LDA (GLDA) (Das et al., 2015) directly generates word embeddings with the Gaussian distribution. The model proposed in Xun et al. (2017) further extends GLDA by modeling topic correlations. MetaLDA (Zhao et al., 2017c; 2018a) is a conjugate topic model that incorporates both document and word meta information. However, in MetaLDA, word embeddings have to be binarized, which can lose useful information. WEI-FTM (Zhao et al., 2017b) is a focused topic model where a topic focuses on a subset of words, informed by word embeddings. To our knowledge, topic hierarchies and sub-topics are not considered in most of the existing word embedding models.

3. The Proposed Model

Based on the PFA framework, WEDTM is a hierarchical model with two major components: one for discovering the inter topic hierarchies and the other for discovering intra topic structures (i.e., sub-topics) informed by word embeddings. The two components are connected by the bottom-layer topics, detailed as follows. Assume that each document j is presented as a word count vector $\boldsymbol{x}_j^{(1)} \in \mathbb{N}_0^V$, where V is the size of the vocabulary; the pre-trained L

[1] https://github.com/ethanhezhao/WEDTM

dimensional real-valued embeddings for each word $v \in \{1, \cdots, V\}$ are stored in a L-dimensional vector $\boldsymbol{f}_v \in \mathbb{R}^L$. Now we consider WEDTM with T hidden layers, where the t-th layer is with K_t topics and k_t is the index of each topic. In the bottom layer ($t = 1$), there are K_1 (local) topics, each of which is associated with S sub-topics. To assist clarity, we split the generative process of the model into three parts, shown as follows:

$$\text{Generating documents} \begin{cases} \boldsymbol{\theta}_j^{(1)} \sim \text{Gam}\left[\boldsymbol{\Phi}^{(2)}\boldsymbol{\theta}_j^{(2)}, p_j^{(2)}/(1-p_j^{(2)})\right], \\ \boldsymbol{\phi}_{k_1}^{(1)} \sim \text{Dir}\left(\boldsymbol{\beta}_{k_1}\right), \\ \boldsymbol{x}_j^{(1)} \sim \text{Pois}\left(\boldsymbol{\Phi}^{(1)}\boldsymbol{\theta}_j^{(1)}\right), \end{cases}$$

$$\text{Inter structure} \begin{cases} \boldsymbol{\theta}_j^{(T)} \sim \text{Gam}\left(\boldsymbol{r}, 1/c_j^{(T+1)}\right), \\ \cdots \\ \boldsymbol{\theta}_j^{(t)} \sim \text{Gam}\left(\boldsymbol{\Phi}^{(t+1)}\boldsymbol{\theta}_j^{(t+1)}, 1/c_j^{(t+1)}\right)(t < T), \\ \boldsymbol{\phi}_{k_t}^{(t)} \sim \text{Dir}\left(\eta_0 \mathbf{1}\right)(t > 1), \\ \cdots \end{cases}$$

$$\text{Intra structure} \begin{cases} \boldsymbol{w}_{k_1}^{<s>} \sim \mathcal{N}\left[\mathbf{0}, \text{diag}(1/\boldsymbol{\sigma}^{<s>})\right], \\ \alpha_{k_1}^{<s>} \sim \text{Gam}(\alpha_0^{<s>}/S, 1/c_0^{<s>}), \\ \beta_{vk_1}^{<s>} \sim \text{Gam}\left(\alpha_{k_1}^{<s>}, e^{\boldsymbol{f}_v^\top \boldsymbol{w}_{k_1}^{<s>}}\right), \\ \beta_{vk_1} := \sum_s^S \beta_{vk_1}^{<s>}, \end{cases}$$

where $^{(t)}$ is the index of the layer that a variable belongs to and $^{<s>}$ is the index of sub-topic s. To complete the model, we impose the following priors on the latent variables:

$$r_{k_T} \sim \text{Gam}(\gamma_0/K_T, 1/c_0),$$
$$\gamma_0 \sim \text{Gam}(a_0, 1/b_0), p_j^{(t)} \sim \text{Beta}(a_0, b_0),$$
$$c_j^{(t)} \sim \text{Gam}(e_0, 1/f_0), \alpha_0^{<s>} \sim \text{Gam}(e_0, 1/f_0),$$
$$c_0^{<s>} \sim \text{Gam}(e_0, 1/f_0), \sigma_l^{<s>} \sim \text{Gam}(a_0, 1/b_0).$$

We first take a look at the bottom layer of the model, i.e., the process of generating the documents, which follows a PFA framework. In this part, WEDTM models the word counts $\boldsymbol{x}_j^{(1)}$ in a document by a Poisson (Pois) distribution and factorizes the Poisson parameters into a product of the factor loadings $\boldsymbol{\Phi}^{(1)} \in \mathbb{R}_+^{V \times K_1}$ and hidden units $\boldsymbol{\theta}_j^{(1)}$. $\boldsymbol{\theta}_j^{(1)}$ is the first-layer latent representation (unnormalized topic weights) of document j, each element of which is drawn from a gamma (Gam) distribution[2]. The k_1-th column of $\boldsymbol{\Phi}^{(1)}$, $\boldsymbol{\phi}_{k_1}^{(1)} \in \mathbb{R}_+^V$ is the word distribution of topic k_1, drawn from a Dirichlet (Dir) distribution. We then explain the component for discovering inter topic hierarchies, which is similar to the structure of GBN (Zhou et al., 2016). Specifically, the shape parameter of $\boldsymbol{\theta}_j^{(1)}$ is factorized into $\boldsymbol{\theta}_j^{(2)}$ and $\boldsymbol{\Phi}^{(2)} \in \mathbb{R}_+^{K_1 \times K_2}$, where $\boldsymbol{\theta}_j^{(2)}$ is the second-layer latent representation of document j and $\boldsymbol{\phi}_{k_2}^{(2)} \in \mathbb{R}_+^{K_1}$ models the correlations between topic k_2 and all the first-layer topics. Note that strictly speaking, k_2 is not a "real" topic as it is not a distribution over words. But it can be interpreted with words by $\boldsymbol{\Phi}^{(1)}\boldsymbol{\phi}_{k_2}^{(2)}$. By repeating this construction, we are able to build a deep structure to discover topic hierarchies.

Now we explain how sub-topics are discovered for the bottom-layer topics with the help of word embeddings. First of all, WEDTM applies individual asymmetric Dirichlet parameters $\boldsymbol{\beta}_{k_1} \in \mathbb{R}_+^V$ for each bottom-layer (local) topic $\boldsymbol{\phi}_{k_1}^{(1)}$. We further construct $\beta_{vk_1} = \sum_s^S \beta_{vk_1}^{<s>}$, where $\beta_{vk_1}^{<s>}$ models how strongly word v is associated with sub-topic s in local topic k_1. For each sub-topic s, we introduce an L-dimensional sub-topic embedding: $\boldsymbol{w}_{k_1}^{<s>} \in \mathbb{R}^L$. As $\beta_{vk_1}^{<s>}$ is gamma distributed, its scale parameter is constructed by the dot product of the embeddings of sub-topic s and word v through the exponential function.

The basic idea of our model is summarized as follows:

1. In terms of sub-topics, we assume each (local, bottom-layer) topic is associated with several sub-topics, in a way that the sub-topics contribute to the prior of the local topic via a sum model (Zhou, 2016). Therefore, if a word dominates in one or more sub-topics, it is likely that the word will still dominate in the local topic. With this construction, a sub-topic is expected to capture one fine-grained thematic aspect of the local topic and each sub-topic can be directly interpreted with words via $\boldsymbol{\beta}_{k_1}^{<s>} \in \mathbb{R}_+^V$.

2. To leverage word embeddings to inform the learning of sub-topics, we introduce the sub-topic embedding for each of them, $\boldsymbol{w}_{k_1}^{<s>}$, which directly interacts with the word embeddings. Therefore, sub-topic embeddings are learned with both the local context of the target corpus and the global information of word embeddings. According to our model construction, the probability density function of β_{vk_1} is the convolution of S covariance-dependent gamma distributions (Zhou, 2016). Therefore, if the sub-topic embeddings of s and word embeddings of v are close, the dot product of them will be large, giving a large expectation of $\beta_{vk_1}^{<s>}$. The large expectation means that v has a large weight in sub-topic s of k. Finally, $\beta_{vk_1}^{<s>}$ further contributes to the local topic's prior β_{vk_1}, informing $\phi_{vk_1}^{(1)}$ of the local topic.

3. It is also noteworthy the special case of WEDTM, where $S = 1$, meaning that there are no sub-topics and each local topic k_1 is associated with one topic embedding vector $\boldsymbol{w}_{k_1}$. Consequently, in WEDTM, there are three latent variables capturing the weights between the words and local topic k_1: $e^{\mathbf{F}^\top \boldsymbol{w}_{k_1}}$ ($\mathbf{F} \in \mathbb{R}^{L \times V}$ is

[2] The first and second parameters of the gamma distribution are the shape and scale respectively.

the embeddings of all the words), $\boldsymbol{\beta}_{k_1}$, and $\boldsymbol{\phi}_{k_1}^{(1)}$, each of which is a vector over words. It is interesting to analyze the connections and differences of them. $e^{\mathbf{F}^\top \boldsymbol{w}_{k_1}}$ is the prior of $\boldsymbol{\beta}_{k_1}$, while $\boldsymbol{\beta}_{k_1}$ is the prior of $\boldsymbol{\phi}_{k_1}^{(1)}$. So $e^{\mathbf{F}^\top \boldsymbol{w}_{k_1}}$ is the closest one to the word embeddings, i.e., the global semantic information, while $\boldsymbol{\phi}_{k_1}^{(1)}$ is the closest one to the data, i.e., the local document context of the target corpus. Therefore, unlike conventional topic models with $\boldsymbol{\phi}_{k_1}^{(1)}$ only, the three variables of WEDTM give three different views to the same topic, from global to local, respectively. We qualitatively show this interesting comparison in Section 5.4.

4. The last but not least, word embeddings in WEDTM can be viewed to serve as the prior/complementary information to assist the learning of the whole model, which is important especially for sparse data.

4. Inference

Unlike many other word embeddings topic models, the fully local conjugacy of WEDTM facilitates the derivation of an effective Gibbs sampling algorithm. As the sampling for the latent variables in the process of generating documents and modeling inter topic structure are similar to GBN, the details can be found in Zhou et al. (2016). Here we focus on the sampling of the latent variables for modeling intra topic structure.

Assume that sampled by Eq. (28) in Appendix B of Zhou et al. (2016), the latent count for the bottom-layer local topics are $x_{vjk_1}^{(1)}$, which counts how many words v in document j are allocated with local topic k_1.

Sample $\beta_{vk_1}^{<s>}$. We first sample:

$$\left(h_{vk_1}^{<1>}, \cdots, h_{vk_1}^{<S>}\right) \sim \text{Mult}\left(h_{vk_1}, \frac{\beta_{vk_1}^{<1>}}{\beta_{vk_1}}, \cdots, \frac{\beta_{vk_1}^{<S>}}{\beta_{vk_1}}\right), \tag{1}$$

where $h_{vk_1} \sim \text{CRT}\left(x_{v\cdot k_1}^{(1)}, \beta_{vk_1}\right)$ (Zhou & Carin, 2015; Zhao et al., 2017a), and $x_{v\cdot k_1}^{(1)} := \sum_j x_{vjk_1}^{(1)}$ [3]. Then:

$$\beta_{vk_1}^{<s>} \sim \frac{\text{Gam}(\alpha_{k_1}^{<s>} + h_{vk_1}^{<s>}, 1)}{e^{-\pi_{vk_1}^{<s>}} + \log \frac{1}{q_{k_1}}}, \tag{2}$$

where $q_{k_1} \sim \text{Beta}(\beta_{\cdot k_1}, x_{\cdot\cdot k_1}^{(1)})$ (Zhao et al., 2018b) and we define $\pi_{vk_1}^{<s>} := \boldsymbol{f}_v^\top \boldsymbol{w}_{k_1}^{<s>}$.

Sample $\alpha_k^{<s>}$. We first sample $g_{vk_1}^{<s>} \sim \text{CRT}\left(h_{vk_1}^{<s>}, \alpha_{k_1}^{<s>}\right)$, then:

$$\alpha_{k_1}^{<s>} \sim \frac{\text{Gam}(\alpha_0^{<s>}/S + g_{\cdot k_1}^{<s>}, 1)}{c_0^{<s>} + \log\left(1 + e^{\pi_{vk_1}^{<s>}} \log \frac{1}{q_{k_1}}\right)}. \tag{3}$$

It is noteworthy that the hierarchical construction on $\alpha_{k_1}^{<s>}$ is closely related to the gamma-negative binomial process and can be considered as a (truncated) gamma process (Zhou & Carin, 2015; Zhou, 2016) with an intrinsic shrinkage mechanism on S. It means that the model is able to automatically learn the number of effective sub-topics.

Sample $\boldsymbol{w}_{k_1}^{<s>}$.

$$\begin{aligned} \boldsymbol{w}_{k_1}^{<s>} &\sim \mathcal{N}(\boldsymbol{\mu}_{k_1}^{<s>}, \boldsymbol{\Sigma}_{k_1}^{<s>}), \\ \boldsymbol{\mu}_{k_1}^{<s>} &= \\ \boldsymbol{\Sigma}_{k_1}^{<s>} & \left[\sum_v^V \left(\frac{h_{vk_1}^{<s>} - \alpha_{k_1}^{<s>}}{2} - \omega_{vk_1}^{<s>} \log\log \frac{1}{q_{k_1}}\right) \boldsymbol{f}_v\right], \\ \boldsymbol{\Sigma}_{k_1}^{<s>} &= \left[\text{diag}(1/\boldsymbol{\sigma}^{<s>}) + \sum_v^V \omega_{vk_1}^{<s>} \boldsymbol{f}_v (\boldsymbol{f}_v)^\top\right]^{-1}, \end{aligned} \tag{4}$$

where $\omega_{vk_1}^{<s>} \sim \text{PG}\left(h_{vk_1}^{<s>} + \alpha_{k_1}^{<s>}, \pi_{vk_1}^{<s>} + \log\log \frac{1}{q_{k_1}}\right)$ and PG denotes the Pólya gamma distribution (Polson et al., 2013). To sample from PG, we use an accurate and efficient approximate sampler in Zhou (2016).

Omitted derivations, details, and the overall algorithm are in the supplementary materials.

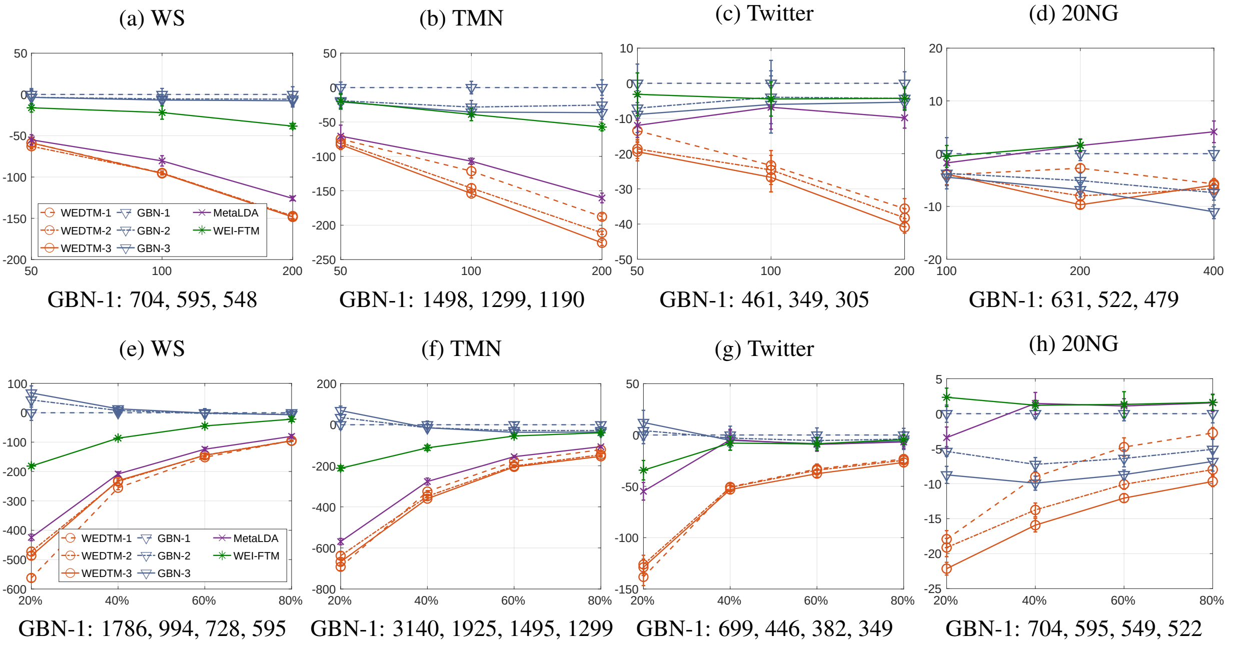
5. Experiments

We evaluate the proposed WEDTM by comparing it with several recent advances including deep topic models and word embedding topic models. The experiments were conducted on four real-world datasets including both regular and sparse texts. We report perplexity, document classification accuracy, and topic coherence scores. We also qualitatively analyze the topic hierarchies and sub-topics.

5.1. Experimental Settings

In the experiments, we used a regular text dataset (20NG) and three sparse text datasets (WS, TMN, Twitter), the details of which are as follows: **1. 20NG**, 20 Newsgroup, consists of 18,774 articles with 20 categories. Following Zhou et al. (2016), we used the 2000 most frequent terms after removing stopwords. The average document length is 76. **2. WS**, Web Snippets, contains 12,237 web search snippets with 8 categories, used by Li et al. (2016); Zhao et al. (2017c;b). The vocabulary contains 10,052 tokens and there are 15 words in one snippet on average. **3. TMN**, Tag My

[3] We hereafter use $\cdot$ of a dimension to denote the sum over that dimension.


(a) WS (b) TMN (c) Twitter (d) 20NG

GBN-1: 704, 595, 548 GBN-1: 1498, 1299, 1190 GBN-1: 461, 349, 305 GBN-1: 631, 522, 479

(e) WS (f) TMN (g) Twitter (h) 20NG

GBN-1: 1786, 994, 728, 595 GBN-1: 3140, 1925, 1495, 1299 GBN-1: 699, 446, 382, 349 GBN-1: 704, 595, 549, 522

Figure 1. (a)-(d): Relative per-heldout-word perplexity[6] (the lower the better) with the varied K_1 and fixed proportion (80%) of training words of each document. (e)-(h): Relative per-heldout-word perplexity[6] with the varied proportion of training words of each document and fixed K_1 (100 on WS, TMN, and Twitter; 200 for 20NG). The error bars indicate the standard deviations of 5 random trials. The number attached to WEDTM and GBN indicates the number of layers (i.e., T) used.

News, consists of 32,597 RSS news snippets from Tag My News with 7 categories, used by Nguyen et al. (2015); Zhao et al. (2017c;b). Each snippet contains a title and a short description. There are 13,370 tokens in the vocabulary and the average length of a snippet is 18. **4. Twitter**, was extracted in 2011 and 2012 microblog tracks at Text REtrieval Conference (TREC)[4] and preprocessed in Yin & Wang (2014). It has 11,109 tweets in total. The vocabulary size is 6,344 and a tweet contains 21 words on average.

We compared WEDTM with: **1. GBN** (Zhou et al., 2016), the state-of-the-art deep topic model. **2. MetaLDA** (Zhao et al., 2017c; 2018a), the state-of-the-art topic model with binary meta information about document and/or word. Word embeddings need to be binarized before used in the model. **3. WEI-FTM** (Zhao et al., 2017b), the state-of-the-art focused topic model that incorporates real-valued word embeddings.

It is noteworthy that GBN was reported (Cong et al., 2017) to have better performance than other deep (hierarchical) topic models such as nHDP (Paisley et al., 2015), DPFA (Gan et al., 2015), and DPFM (Henao et al., 2015). MetaLDA and WEI-FTM were reported to perform better than other word embedding topic models including WF-LDA (Petterson et al., 2010) and GPUDMM (Li et al., 2016) as well as short text topic models like PTM (Zuo et al., 2016). Therefore, we considered the three above competitors to WEDTM.

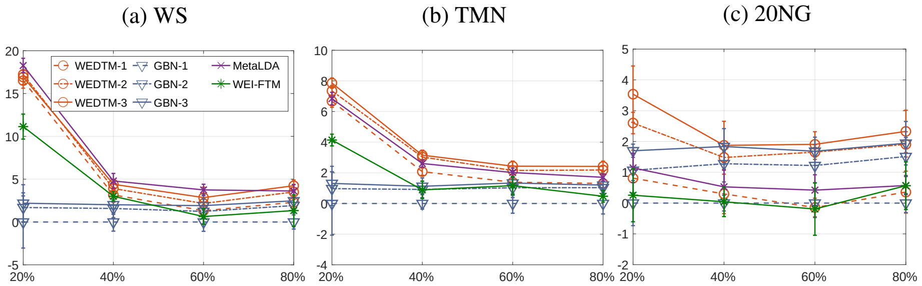
Originally MetaLDA (when no document meta information is provided) and WEI-FTM follow the LDA framework, where the topic distribution for document j is $\boldsymbol{\theta_j} \sim \mathrm{Dir}(\alpha_0 \mathbf{1})$ and α_0 is a hyperparameter (usually set to 0.1). For a fair comparison, we replaced this part with the PFA framework with the gamma-negative binomial process (Zhou & Carin, 2015), which is equivalent to GBN when $T = 1$ and closely related to the hierarchical Dirichlet Process LDA (HDPLDA) (Teh et al., 2012).

For all the models, we used 50-dimensional GloVe word embeddings pre-trained on Wikipedia[5]. Except for MetaLDA, where we followed the paper to binarise the word embeddings, the other three models used the original real-valued embeddings. The hyperparameter settings we used for WEDTM and GBN are $a_0 = b_0 = 0.01, e_0 = f_0 = 1.0, \eta_0 = 0.05$. For MetaLDA and WEI-FTM, we collected 1000 MCMC samples after 1000 burnins; for GBN and WEDTM, we collected 1000 for $T = 1$ and 500 for $T > 1$ MCMC samples after 1000 for $T = 1$ and 500 for $T > 1$ burnins, to estimate the posterior mean. Due to the shrinkage effect of WEDTM on S, discussed in Section 4, we set $S = 5$ which is large enough for all the topics.

[4] http://trec.nist.gov/data/microblog.html

[5] https://nlp.stanford.edu/projects/glove/

WEDTM


(a) WS (b) TMN (c) 20NG

GBN-1: 66.6, 80.9, 83.1, 82.2 GBN-1: 72.0, 77.9, 78.8, 79.4 GBN-1: 64.5, 68.3, 69.6, 69.9

Figure 2. Relative document classification accuracy[6] (%) on WS, TMN, and 20NG with the varied proportion of training words of each training document. The results with $K_1 = 100$ on WS and TMN, $K_1 = 200$ on 20NG are reported.

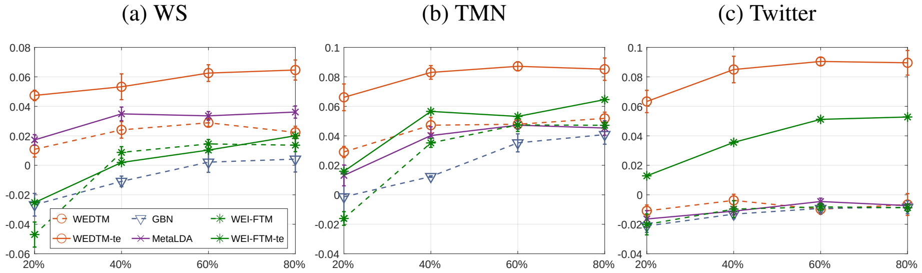

(a) WS (b) TMN (c) Twitter

Figure 3. Topic coherence (NPMI, the higher the better) on WS, TMN, and Twitter with the varied proportion of training words of each document. The results with $K_1 = 100$ are reported. For WEDTM and WEI-FTM, the top words of a topic are generated by ranking the word distribution and the dot product of topic and word embeddings (denoted "-te").

5.2. Perplexity

Perplexity is a measure that is widely used (Wallach et al., 2009) to evaluate the modeling accuracy of topic models. Here we randomly chose a certain proportion of the word tokens in each document as training and used the remaining ones to calculate per-heldout-word perplexity. Figure 1 shows the relative perplexity[6] results of all the models on all the datasets, where we varied the number of bottom-layer topics as well as the proportion of training words. The proposed WEDTM performs significantly better than the others, especially on sparse data. There are several interesting remarks of the results: (1) The perplexity advantage of WEDTM over GBN becomes obvious when the corpus becomes sparse (e.g., WS/TMN/Twitter V.S. 20NG and 20% V.S. 80% training words). It shows that using word embeddings as the prior information benefits the model. (2) In general, increasing the depth of the model leads to better perplexity. However, when the data are too sparse (e.g. WS with 20% training words), the single-layer WEDTM and GBN perform better than their multi-layer counterparts. (3) Although MetaLDA and WEI-FTM leverage word embeddings as well, the proposed WEDTM outperforms them significantly. Perhaps the way that WEDTM incorporates word embeddings is more effective.

5.3. Document Classification

We consider the multi-class classification task for predicting the categories for test documents to evaluate the quality of the latent document representation (unnormalized topic weights) extracted by these models.[7] In this experiment, following Zhou et al. (2016), we ran the topic models on the training documents and trained a L_2 regularized logistic regression using the LIBLINEAR package (Fan et al., 2008) with the latent representation $\boldsymbol{\theta}_j^{(1)}$ as features. After training, we used the trained topic models to extract the latent representations of the test documents and the trained logistic regression to predict the categories. For all the datasets, we randomly selected 80% documents for training and used the remaining 20% for testing. Figure 2 shows the relative document classification accuracy[6] results for all the models. It can be observed that with word embeddings, WEDTM outperforms GBN significantly, the best on TMN and 20NG, and the second-best on WS. Again, we see a similar phe-

[6] We subtracted the score of GBN with only one layer (GBN-1) from the score of each model. The lines plot the differences. So GBN-1 is the horizontal line on "0". The absolute score of GBN-1 is given below each figure.

[7] The results of Twitter are not reported because each document of it is associated with multiple categories.

nomenon: word embeddings help more on the sparser data and increasing the network depth improves the accuracy.

5.4. Topic Coherence

Topic coherence is another popular evaluation of topic models (Zuo et al., 2016; Zhao et al., 2017b;b). It measures the semantic coherence in the most significant words (top words) in a topic. Here we used the Normalized Pointwise Mutual Information (NPMI) (Aletras & Stevenson, 2013; Lau et al., 2014) to calculate topic coherence score of the top 10 words of each topic and report the average score of all the topics.[8]

To compare with the other models, in this experiment, we set $S = 1$ for WEDTM. Recall that in WEDTM, from global to local, there are three ways to interpret a topic. Here we evaluate NPMI for two of them: $e^{\mathbf{F}^\top \boldsymbol{w}_{k_1}}$ and $\boldsymbol{\phi}_{k_1}^{(1)}$. Figure 3 shows the NPMI scores for all the models on WS, TMN, and Twitter. It is not surprising to see that the top words generated by $e^{\mathbf{F}^\top \boldsymbol{w}_{k_1}}$ in WEDTM always gain the highest NPMI scores, meaning that the topics are more coherent. This is because the topic embeddings in WEDTM directly interact with word embeddings. Moreover, if we just compare the topics generated by $\boldsymbol{\phi}_{k_1}^{(1)}$, WEDTM also gives more coherent topics than the other models. This demonstrates that the proposed model is able to discover more interpretable topics.

5.5. Qualitative Analysis

As one of the most appealing properties of WEDTM is its interpretability, we conducted the extensive qualitative evaluation of the quality of the topics discovered by WEDTM, including topic embeddings, sub-topics, and topic hierarchies. More qualitative analysis including topic hierarchy visualization and synthetic document generation is shown in the supplementary materials.

Demonstration of topic embeddings: We demonstrate that WEDTM discovers more coherent topics by comparing with those of GBN in Table 2. Here we set $S = 1$ as well. This demonstration further explains the numerical results in Figure 3. It is also interesting to compare the local interpretation ($\boldsymbol{\phi}_k^{(1)}$) and global interpretation (topic embeddings) of the same topic in WEDTM. For example, in the fifth set, the local interpretation (5.b) is about "networks and security," while the global interpretation (5.c) generalizes it with more general words related to "communications." We can also observe that although the local interpretation of WEDTM is not as close to word embeddings as the global interpretation, as informed by the global interpretation, the local interpretation of WEDTM's topics is still considerably more coherent than those in GBN.

Demonstration of sub-topics: In Figure 4, We show the sub-topics discovered by WEDTM for the topics used as examples at the beginning of the paper (Table 1). It can be observed that the intra topic structures with sub-topics clearly help to explain the local topics. For example, WEDTM successfully splits Topic 1 into sub-topics related to "journal" and "biology," and Topic 2 into "music" and "sports". Moreover, with the help of word embeddings, WEDTM discovers general sub-topics for specific topics. For example, Topic 3 and 4 are more interpretable with the sub-topics of "singer" and "game" respectively. The experiment also empirically demonstrates the shrinkage mechanism of the model: for most topics, the effective sub-topics are less than the maximum number $S = 5$.

Demonstration of topic hierarchies: Figure 5 shows an example that jointly demonstrates the inter and intra structures of WEDTM. The tree is a cluster of topics related to "health," where the topic hierarchies are discovered by ranking $\{\boldsymbol{\Phi}^{(t)}\}_t$, the leaf nodes are the topics in the bottom layer, and each bottom-layer topic is associated with a set of sub-topics. In WEDTM, the inter topic structures are revealed in the form of topic hierarchies while the intra topic structures are revealed in the form of sub-topics. Combining the two kinds of topic structures in this way gives a better view of the target corpus, which may further benefit other text analysis tasks.

6. Conclusion

In this paper, we have proposed WEDTM, a deep topic model that leverages word embeddings to discover inter topic structures with topic hierarchies and intra topic structures with sub-topics. Moreover, with the introduction to sub-topic embeddings, each sub-topic can be informed by the global information in word embeddings, so as to discover a fine-grained thematic aspect of a local topic. With topic embeddings, WEDTM provides different views to a topic, from global to local, which further improves the interpretability of the model. As a fully conjugate model, the inference of WEDTM can be done by a straightforward Gibbs sampling algorithm. Extensive experiments have shown that WEDTM achieves the state-of-the-art performance on perplexity, document classification, and topic quality. In addition, with topic hierarchies, sub-topics, and topic embeddings, the model can discover more interpretable structured topics, which helps to get better understandings of text data. Given the local conjugacy, it is possible to derive more scalable inference algorithms for WEDTM, such as stochastic variational inference and stochastic gradient MCMC, which is a good subject for future work.

[8] We used the Palmetto package with a large Wikipedia dump to compute NPMI (`http://palmetto.aksw.org`).

Table 2. Top 10 words of five sets of example topics on the WS dataset. Each set contains the top words of 3 topics: topic 'a' is generated by $\phi_k^{(1)}$ in GBN-3; topic 'b' is generated by $\phi_k^{(1)}$ in WEDTM-3; topic 'c' is generated by $e^{\mathbf{F}^\top \boldsymbol{w}_{k1}}$ in WEDTM-3. Topic 'a' and 'b' are matched by the Hellinger distance of $\phi_k^{(1)}$. Topic 'b' and 'c' are different ways of interpreting one topic in WEDTM.

Topic	Index	Top 10 words	NPMI
1	a	engine car buying home diesel selling fuel automobile violin jet	-0.055
	b	engine motor diesel fuel gasoline jet electric engines gas technology	0.202
	c	engine diesel engines gasoline steam electric fuel propulsion motors combustion	0.224
2	a	party labor democratic political socialist movement union social news australian	0.168
	b	party political communist democratic socialist labor republican parties conservative leader	0.188
	c	party democratic communist labour liberal socialist conservative opposition elections republican	0.219
3	a	cancer lung tobacco intelligence artificial information health symptoms smoking treatment	-0.006
	b	cancer lung tobacco information health smoking treatment gov research symptoms	0.050
	c	cancer breast diabetes pulmonary cancers patients asthma cardiovascular cholesterol obesity	0.050
4	a	oscar academy awards swimming award winners swim oscars nominations picture	0.020
	b	art awards oscar academy gallery museum surrealism sculpture picasso arts	0.076
	c	paintings awards award art museum gallery sculpture painting picasso portrait	0.087
5	a	security computer network nuclear weapons networking spam virus spyware national	0.059
	b	security network wireless access networks spam spyware networking national computer	0.061
	c	wireless internet networks devices phone broadband users network wi-fi providers	0.143

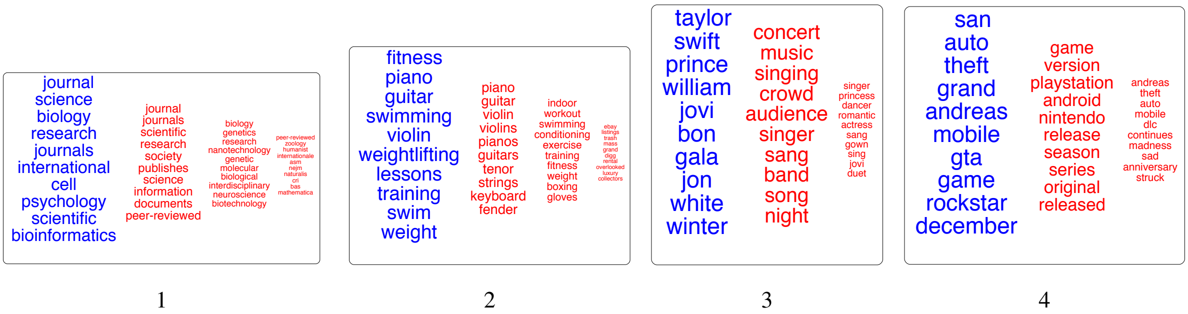

journal science biology research journals international cell psychology scientific bioinformatics
journal journals scientific research society publishes science information documents peer-reviewed
biology genetics research nanotechnology genetic molecular biological interdisciplinary neuroscience biotechnology
peer-reviewed zoology humanist internationale asm nejm naturalis cri bas mathematica
fitness piano guitar swimming violin weightlifting lessons training swim weight
piano guitar violin violins pianos guitars tenor strings keyboard fender
indoor workout swimming conditioning exercise training fitness weight boxing gloves
ebay listings trash mass grand digg rental overlooked luxury collectors
taylor swift prince william jovi bon gala jon white winter
concert music singing crowd audience singer sang band song night
singer princess dancer romantic actress sang gown sing jovi duet
san auto theft grand andreas mobile gta game rockstar december
game version playstation android nintendo release season series original released
andreas theft auto mobile dlc continues madness sad anniversary struck
1
2
3
4

Figure 4. The sub-topics (red) of the example topics (blue). Larger font size indicates larger weight ($\sum_v^V \beta_{vk}^{<s>}$) of a sub-topic to the local topic. We set $S = 5$ and trimmed off the sub-topics with extreme small weights.

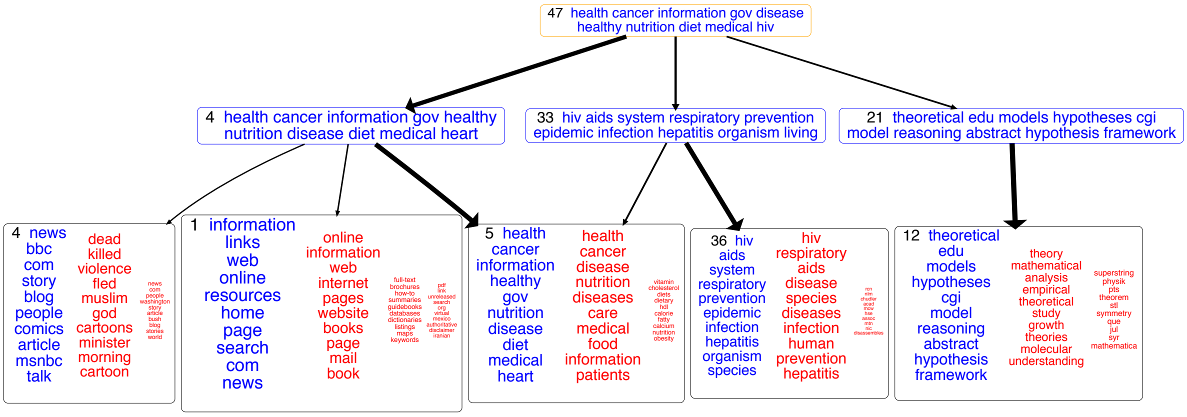

47 health cancer information gov disease healthy nutrition diet medical hiv
4 health cancer information gov healthy nutrition disease diet medical heart
33 hiv aids system respiratory prevention epidemic infection hepatitis organism living
21 theoretical edu models hypotheses cgi model reasoning abstract hypothesis framework
4 news bbc com story blog people comics article msnbc talk
dead killed violence fled muslim god cartoons minister morning cartoon
news com people washington story article bush blog stories world
1 information links web online resources home page search com news
online information web internet pages website books page mail book
full-text brochures how-to summaries guidebooks databases dictionaries listings maps keywords
pdf link unreleased search org virtual mexico authoritative disclaimer iranian
5 health cancer information healthy gov nutrition disease diet medical heart
health cancer disease nutrition diseases care medical food information patients
vitamin cholesterol diets dietary hdl calorie fatty calcium nutrition obesity
36 hiv aids system respiratory prevention epidemic infection hepatitis organism species
hiv respiratory aids disease species diseases infection human prevention hepatitis
rcn nlm chudler acad mcw hse assoc mtn nic disassembles
12 theoretical edu models hypotheses cgi model reasoning abstract hypothesis framework
theory mathematical analysis empirical theoretical study growth theories molecular understanding
superstring physik pts theorem stl symmetry que jul syr mathematica

Figure 5. One example sub-tree of the topic hierarchy discovered by WEDTM on the WS dataset with $K_1 = 50$ and $S = 5$. The tree is generated in the same way to Zhou et al. (2016). A line from node k_t at layer t to node k_{t-1} at layer $t-1$ indicates that $\phi_{k_{t-1}k_t}^{(t)} > 1.5/K_{t-1}$ and its width indicates the value of $\phi_{k_{t-1}k_t}^{(t)}$ (i.e. topic correlation strength). The outside border of the text box is colored as orange, blue, or black if the node is at layer three, two, or one, respectively. For the leaf nodes, sub-topics are shown in the same way to Figure 4.

References

Ahmed, A., Hong, L., and Smola, A. Nested Chinese restaurant franchise process: Applications to user tracking and document modeling. In *ICML*, 2013.

Aletras, N. and Stevenson, M. Evaluating topic coherence using distributional semantics. In *Proc. of the 10th International Conference on Computational Semantics*, pp. 13–22, 2013.

Blei, D., Griffiths, T., and Jordan, M. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *Journal of the ACM (JACM)*, 57(2): 7, 2010.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. Enriching word vectors with subword information. *TACL*, 5: 135–146, 2017.

Cong, Y., Chen, B., Liu, H., and Zhou, M. Deep latent Dirichlet allocation with topic-layer-adaptive stochastic gradient Riemannian MCMC. In *ICML*, pp. 864–873, 2017.

Das, R., Zaheer, M., and Dyer, C. Gaussian LDA for topic models with word embeddings. In *ACL*, pp. 795–804, 2015.

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. LIBLINEAR: A library for large linear classification. *JMLR*, 9:1871–1874, 2008.

Gan, Z., Chen, C., Henao, R., Carlson, D., and Carin, L. Scalable deep Poisson factor analysis for topic modeling. In *ICML*, pp. 1823–1832, 2015.

Henao, R., Gan, Z., Lu, J., and Carin, L. Deep Poisson factor modeling. In *NIPS*, pp. 2800–2808, 2015.

Kim, J. H., Kim, D., Kim, S., and Oh, A. Modeling topic hierarchies with the recursive Chinese restaurant process. In *CIKM*, pp. 783–792. ACM, 2012.

Lau, J. H., Newman, D., and Baldwin, T. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *EACL*, pp. 530–539, 2014.

Li, C., Wang, H., Zhang, Z., Sun, A., and Ma, Z. Topic modeling for short texts with auxiliary word embeddings. In *SIGIR*, pp. 165–174, 2016.

Li, W. and McCallum, A. Pachinko allocation: DAG-structured mixture models of topic correlations. In *ICML*, pp. 577–584, 2006.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed representations of words and phrases and their compositionally. In *NIPS*, pp. 3111–3119, 2013.

Nguyen, D. Q., Billingsley, R., Du, L., and Johnson, M. Improving topic models with latent feature word representations. *TACL*, 3:299–313, 2015.

Paisley, J., Wang, C., Blei, D., and Jordan, M. Nested hierarchical Dirichlet processes. *TPAMI*, 37(2):256–270, 2015.

Pennington, J., Socher, R., and Manning, C. GloVe: Global vectors for word representation. In *EMNLP*, pp. 1532–1543, 2014.

Petterson, J., Buntine, W., Narayanamurthy, S. M., Caetano, T. S., and Smola, A. J. Word features for Latent Dirichlet Allocation. In *NIPS*, pp. 1921–1929, 2010.

Polson, N., Scott, J., and Windle, J. Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American Statistical Association*, 108(504): 1339–1349, 2013.

Teh, Y. W., Jordan, M., Beal, M., and Blei, D. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2012.

Wallach, H. M., Mimno, D. M., and McCallum, A. Rethinking LDA: Why priors matter. In *NIPS*, pp. 1973–1981, 2009.

Xun, G., Li, Y., Zhao, W. X., Gao, J., and Zhang, A. A correlated topic model using word embeddings. In *IJCAI*, pp. 4207–4213, 2017.

Yin, J. and Wang, J. A Dirichlet multinomial mixture model-based approach for short text clustering. In *SIGKDD*, pp. 233–242. ACM, 2014.

Zhao, H., Du, L., and Buntine, W. Leveraging node attributes for incomplete relational data. In *ICML*, pp. 4072–4081, 2017a.

Zhao, H., Du, L., and Buntine, W. A word embeddings informed focused topic model. In *ACML*, pp. 423–438, 2017b.

Zhao, H., Du, L., Buntine, W., and Liu, G. MetaLDA: A topic model that efficiently incorporates meta information. In *ICDM*, pp. 635–644, 2017c.

Zhao, H., Du, L., Buntine, W., and Liu, G. Leveraging external information in topic modelling. *Knowledge and Information Systems*, pp. 1–33, 2018a.

Zhao, H., Rai, P., Du, L., and Buntine, W. Bayesian multi-label learning with sparse features and labels, and label co-occurrences. In *AISTATS*, pp. 1943–1951, 2018b.

Zhou, M. Softplus regressions and convex polytopes. *arXiv preprint arXiv:1608.06383*, 2016.

Zhou, M. Nonparametric Bayesian negative binomial factor analysis. *Bayesian Analysis*, 2018.

Zhou, M. and Carin, L. Negative binomial process count and mixture modeling. *TPAMI*, 37(2):307–320, 2015.

Zhou, M., Hannah, L., Dunson, D. B., and Carin, L. Beta-negative binomial process and Poisson factor analysis. In *AISTATS*, pp. 1462–1471, 2012.

Zhou, M., Cong, Y., and Chen, B. Augmentable gamma belief networks. *JMLR*, 17(163):1–44, 2016.

Zuo, Y., Wu, J., Zhang, H., Lin, H., Wang, F., Xu, K., and Xiong, H. Topic modeling of short texts: A pseudo-document view. In *SIGKDD*, pp. 2105–2114, 2016.

Supplementary Material for "Inter and Intra Topic Structure Learning with Word Embeddings"

He Zhao [1] Lan Du [1] Wray Buntine [1] Mingyaun Zhou [2]

1. Inference Details

Recall that the data is the word count vector $\boldsymbol{x}_j^{(1)}$ of document j. Given the PFA framework, we can apply the collapsed Gibbs sampling to sample the bottom-layer topic for the i-th word in j, v_{ji}, similar to Eq. (28) in Appendix B of Zhou et al. (2016), as follows:

$$P(z_{ji} = k_1) \propto \frac{\beta_{vk_1} + x_{v_{ji} \cdot k_1}^{(1)^{-ji}}}{\beta_{\cdot k_1} + x_{\cdot\cdot k_1}^{(1)^{-ji}}} \left(x_{\cdot jk_1}^{(1)^{-ji}} + \boldsymbol{\phi}_{k_1:}^{(2)} \boldsymbol{\theta}_j^{(2)} \right) \tag{1}$$

where z_{ji} is the topic index for v_{ji} and $x_{vjk}^{(1)} := \sum_i \delta(v_{ji} = v, z_{ji} = k_1)$ counts the number of times that term v appears in document j; we use x^{-ji} to denote the count x calculated without considering word i in document j.

Given the latent counts $x_{v\cdot k_1}^{(1)}$, the multinomial likelihood of $\boldsymbol{\phi}_{k_1}^{(1)}$ is proportional to $\left(\phi_{vk_1}^{(1)}\right)^{x_{v\cdot k_1}^{(1)}}$. Due to the Dirichlet-multinomial conjugacy, the joint likelihood of $\boldsymbol{\beta}_{k_1}$ is computed as:

$$\mathcal{L}\left(\beta_{vk_1}\right) \propto \frac{\Gamma(\beta_{\cdot k_1})}{\Gamma(\beta_{k_1 \cdot} + x_{\cdot\cdot k_1}^{(1)})} \prod_v^V \frac{\Gamma(\beta_{vk_1} + x_{v\cdot k_1}^{(1)})}{\Gamma(\beta_{vk_1})}. \tag{2}$$

By introducing an auxiliary beta distributed variable:

$$q_{k_1} \sim \text{Beta}(\beta_{\cdot k_1}, x_{\cdot\cdot k_1}^{(1)}), \tag{3}$$

we can transform the first gamma ratio in Eq. (2) to $(q_{k_1})^{\beta_{\cdot k_1}}$ (Zhao et al., 2018). After this augmentation, one can show that the likelihood of $\beta_{k_1,v}$ is proportional to the negative binomial distribution.

Recall that $\beta_{vk_1} = \sum_s^S \beta_{vk_1}^{<s>}$ and the shape parameter of $\beta_{vk_1}^{<s>}$, $\alpha_{k_1}^{<s>}$ is drawn from hierarchical gamma. This construction is closely related to the gamma-negative binomial process (Zhou & Carin, 2015; Zhou, 2016), which enables the model to automatically determine the number of effective sub-topics.

Furthermore, we introduce another auxiliary variable:

$$h_{vk_1} \sim \text{CRT}\left(x_{v\cdot k_1}^{(1)}, \beta_{vk_1}\right), \tag{4}$$

where $h \sim \text{CRT}(n, r)$ stands for the Chinese Restaurant Table distribution (Zhou & Carin, 2015) that generates the number of tables h seated by n customers in a Chinese restaurant process with the concentration parameter r (Wray & Marcus, 2012). Given h_{vk_1}, the second gamma ratio can be augmented as $(\beta_{vk_1})^{h_{vk_1}}$. Finally, with the two auxiliary variables, Eq. (2) can be written as:

$$\mathcal{L}\left(\beta_{vk_1}, q_{k_1}, h_{vk_1}\right) \propto (q_{k_1})^{\beta_{vk_1}} (\beta_{vk_1})^{h_{vk_1}}. \tag{5}$$

Sample $\beta_{vk_1}^{<s>}$. Given the table counts h_{vk_1} in Eq. (5), we can sample the counts $h_{vk_1}^{<s>}$ for each sub-topic s as follows:

$$\left(h_{vk_1}^{<1>}, \cdots, h_{vk_1}^{<S>}\right) \sim \text{Mult}\left(h_{vk_1}, \frac{\beta_{vk_1}^{<1>}}{\beta_{vk_1}}, \cdots, \frac{\beta_{vk_1}^{<S>}}{\beta_{vk_1}}\right). \tag{6}$$

Given $h_{vk_1}^{<s>}$, we can sample $\beta_{vk_1}^{<s>}$ as:

$$\beta_{vk_1}^{<s>} \sim \frac{\text{Gam}(\alpha_{k_1}^{<s>} + h_{vk_1}^{<s>}, 1)}{e^{-\pi_{vk_1}^{<s>}} + \log \frac{1}{q_{k_1}}}, \tag{7}$$

where we define

$$\pi_{vk_1}^{<s>} := \boldsymbol{f}_v^\top \boldsymbol{w}_{k_1}^{<s>}. \tag{8}$$

Sample $\alpha_{k_1}^{<s>}$. By integrating $\beta_{vk_1}^{<s>}$ out, we sample $\alpha_{k_1}^{<s>}$ as:

$$\alpha_{k_1}^{<s>} \sim \frac{\text{Gam}(\alpha_0^{<s>}/S + g_{\cdot k_1}^{<s>}, 1)}{c_0^{<s>} + \log\left(1 + e^{\pi_{vk_1}^{<s>}} \log \frac{1}{q_{k_1}}\right)}, \tag{9}$$

where

$$g_{\cdot k_1}^{<s>} := \sum_v^V g_{vk_1}^{<s>}, \tag{10}$$

$$g_{vk_1}^{<s>} \sim \text{CRT}\left(h_{vk_1}^{<s>}, \alpha_{k_1}^{<s>}\right). \tag{11}$$

According to the gamma-gamma conjugacy, $\alpha_0^{<s>}$ and $c_0^{<s>}$ can be sampled in a similar way.

[1] Faculty of Information Technology, Monash University, Australia [2] McCombs School of Business, University of Texas at Austin. Correspondence to: Lan Du <lan.du@monash.edu>, Mingyuan Zhou <mingyuan.zhou@mccombs.utexas.edu>.

Proceedings of the 35th International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

Sample $\boldsymbol{w}_{k_1}^{<s>}$. After integrating $\beta_{vk_1}^{<s>}$ out and ignoring unrelated terms, the joint likelihood related to $\boldsymbol{w}_{k_1}^{<s>}$ is proportional to:

$$\mathcal{L}\left(\pi_{vk_1}^{<s>}\right) \propto \frac{\left(e^{\pi_{vk_1}^{<s>}+\log\log\frac{1}{q_{k_1}}}\right)^{h_{vk_1}^{<s>}}}{\left(1+e^{\pi_{vk_1}^{<s>}+\log\log\frac{1}{q_{k_1}}}\right)^{\alpha_{k_1}^{<s>}+h_{vk_1}^{<s>}}}. \tag{12}$$

The above likelihood can be augmented by an auxiliary variable: $\omega_{vk_1}^{<s>} \sim \mathrm{PG}(1,0)$, where PG denotes the Pólya gamma distribution (Polson et al., 2013). Given $\omega_{vk_1}^{<s>}$, we get:

$$\mathcal{L}\left(\pi_{vk_1}^{<s>}, \omega_{vk_1}^{<s>}\right) \propto e^{\frac{h_{vk_1}^{<s>}-\alpha_{k_1}^{<s>}}{2}\pi_{vk_1}^{<s>}} e^{-\frac{\omega_{vk_1}^{<s>}}{2}\pi_{vk_1}^{<s>2}}. \tag{13}$$

The above likelihood results in the normal likelihood of $\boldsymbol{w}_{k_1}^{<s>}$. Therefore, we sample it from a multi-variate normal distribution as follows:

$$\begin{aligned} \boldsymbol{w}_{k_1}^{<s>} &\sim \mathcal{N}(\boldsymbol{\mu}_{k_1}^{<s>}, \boldsymbol{\Sigma}_{k_1}^{<s>}), \\ \boldsymbol{\mu}_{k_1}^{<s>} &= \\ \boldsymbol{\Sigma}_{k_1}^{<s>} &\left[\sum_{v}^{V}\left(\frac{h_{vk_1}^{<s>}-\alpha_{k_1}^{<s>}}{2} - \omega_{vk_1}^{<s>}\log\log\frac{1}{q_{k_1}}\right)\boldsymbol{f}_v\right], \\ \boldsymbol{\Sigma}_{k_1}^{<s>} &= \left[\mathrm{diag}(1/\boldsymbol{\sigma}^{<s>}) + \sum_{v}^{V}\omega_{vk_1}^{<s>}\boldsymbol{f}_v(\boldsymbol{f}_v)^{\top}\right]^{-1}, \end{aligned} \tag{14}$$

where $\boldsymbol{\mu}_{k_1}^{<s>} \in \mathbb{R}^{L}$ and $\boldsymbol{\Sigma}_{k_1}^{<s>} \in \mathbb{R}^{L\times L}$.

According to (Polson et al., 2013), we can sample

$$\omega_{vk_1}^{<s>} \sim \mathrm{PG}\left(h_{vk_1}^{<s>}+\alpha_{k_1}^{<s>}, \pi_{vk_1}^{<s>}+\log\log\frac{1}{q_{k_1}}\right). \tag{15}$$

To sample from the Pólya gamma distribution, we use an accurate and efficient approximate sampler in Zhou (2016).

Finally, $\boldsymbol{\sigma}^{(s)}$ can be sampled from its gamma posterior.

The inference algorithm is shown in Figure 1.

2. Visualization of the Discovered Topic Hierarchies and Sub-topics

Figure 1-9 show the topic hierarchies discovered by WEDTM on WS, TMN, and Twitter respectively.

3. Generating Synthetic Documents

Below we provide several synthetic documents generated from WEDTM, following the GBN paper (Zhou et al., 2016). Given trained $\{\boldsymbol{\Phi}^{(t)}\}_t$, we used the generative model shown in Figure 1 in the main paper to generate a simulated topic weights $\boldsymbol{\theta}_j^{(1)}$. We show the top words ranked according to $\boldsymbol{\Phi}^{(1)}\boldsymbol{\theta}_j^{(1)}$. Below are some example synthetic documents generated in this manner with WEDTM trained on the TMN dataset. The generated documents are clearly interpretable.

1. study drug cancer risk health people heart women disease patients drugs researchers children brain found finds kids high doctors suggests research medical surgery food diabetes treatment blood years report men young shows year time scientists mets age linked yankees coli long care tuesday good hospital early prostate breast fda developing common adults monday outbreak older weight lower parents problems taking day studies virus aids babies life diet thursday loss levels higher wednesday home evidence make trial tests effective death sox therapy pregnancy experts prevent low patient run pressure pain alzheimer game flu type win number treat start season obesity symptoms

2. mets yankees game season sox win run baseball nfl league hit home time team red inning players day back beat phillies start pitcher innings night runs rangers giants major year games rays big manager york victory indians coach left past boston make draft dodgers hits good pitch hitter career bay tigers blue list play braves fans marlins tuesday years disabled high cubs thursday saturday lockout chicago week nationals top wednesday lead white jays twins streak los days philadelphia field long texas josh pitched ninth friday sunday francisco homer lineup put young college football times roundup pitching cleveland loss sports angeles

3. japan nuclear earthquake plant power tsunami crisis japanese oil quake radiation stocks disaster tokyo prices world friday tuesday fukushima investors hit crippled energy reactor water march monday safety thursday plants economic reactors market government week wednesday year country devastating wall high workers radioactive concerns worst street officials percent companies damage electric report massive fears earnings damaged economy impact food levels agency operator global daiichi plans markets stock chernobyl sales coast growth rise higher states atomic fuel united risk stricken health crude supply dollar low strong demand level recovery day magnitude quarter shares fell struck tepco billion commodities experts gains relief

4. wedding royal prince kate william middleton london queen britain ireland british irish friday eliza-

WEDTM

```
Require: $\{\boldsymbol{x}_j^{(1)}\}_j, T, S, \{K_t\}_t, a_0, b_0, e_0, f_0, \eta_0, MaxIteration$
Ensure: The sampled value for all the latent variables
Randomly initialise all the latent variables according to the generative process;
for $t \leftarrow 1$ to $T$ do
    for $iter \leftarrow 1$ to $MaxIteration$ do
        if $t = 1$ then
            $/ *$ Generating documents $* /$
            Sample the bottom-layer topics by Eq. (1); Calculate $\{x_{vjk_1}^{(1)}\}_{v,j,k_1}$;
        end if
        $/ *$ Inter topic structure $* /$
        Do the upward-downward Gibbs sampler
        (Algorithm 1 in Zhou et al. (2016)) for $\{\boldsymbol{\theta}_j^{(t)}, \boldsymbol{c}_j^{(t)}, \boldsymbol{p}_j^{(t)}\}_{t,j}, \{\boldsymbol{\Phi}^{(t)}\}_t, \boldsymbol{r}$;
        if $t = 1$ then
            $/ *$ Intra topic structure $* /$
            Sample $\{q_{k_1}\}_{k_1}$ by Eq. (3); Sample $\{h_{vk_1}\}_{v,k_1}$ by Eq. (4);
            for $s \leftarrow 1$ to $S$ do
                Sample $\{h_{vk_1}^{<s>}\}_{v,k_1}$ by Eq. (6);
                Sample $\{g_{vk_1}^{<s>}\}_{v,k_1}$ by Eq. (11); Calculate $\{g_{\cdot k_1}^{<s>}\}_{k_1}$ by Eq. (10);
                Sample $\alpha_0^{<s>}$ and $c_0^{<s>}$ from their gamma posterior;
                Sample $\{\alpha_{k_1}^{<s>}\}_{k_1}$ by Eq. (9);
                Sample $\{\omega_{vk_1}^{<s>}\}_{v,k_1}$ by Eq. (15); Sample $\boldsymbol{\sigma}^{<s>}$ from its gamma posterior;
                Sample $\{\boldsymbol{w}_{k_1}^{<s>}\}_{k_1}$ by Eq. (14);
                Calculate $\{\pi_{vk_1}^{<s>}\}_{v,k_1}$ by Eq. (8);
                Sample $\{\beta_{vk_1}^{<s>}\}_{v,k_1}$ by Eq. (7);
            end for
            Calculate $\{\beta_{vk_1}\}_{v,k_1}$;
        end if
    end for
end for
```

Figure 1. Gibbs sampling algorithm for WEDTM

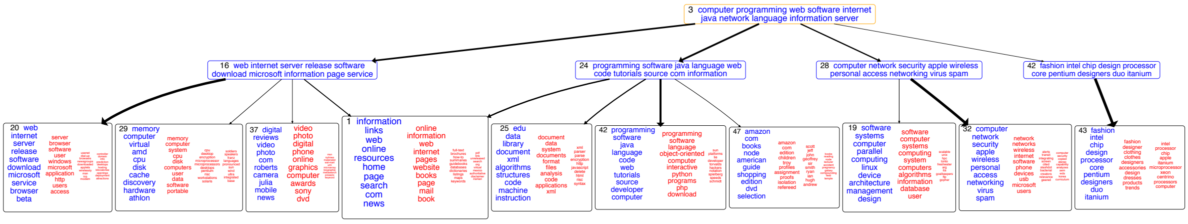

3 computer programming web software internet java network language information server
16 web internet server release software download microsoft information page service
24 programming software java language web code tutorials source com information
28 computer network security apple wireless personal access networking virus spam
42 fashion intel chip design processor core pentium designers duo itanium
20 web
29 memory
37 digital
1 information
25 edu
42 programming
47 amazon
19 software
32 computer
43 fashion

Figure 2. Analogous plots to Figure 6 in the main paper for a tree about "computers" on the WS dataset.

beth designer princess april couple week palace people visit idol american day fashion world made abbey art bride marriage diana duchess secret time honeymoon photo king dress dinner famous sarah cake back media lady buckingham year ring show photos hat coverage wednesday party saturday mcqueen month morning pop watch guests charlie days sheen westminster future kardashian star television night work crown tribute ago duke officials worn check years pre kim maya wear final site met ceremony music tonight museum nuptials harry ferguson pounds royals trip tuesday turned

5. apple sony mobile data ipad company corp network phone billion software google iphone computer mil-

WEDTM

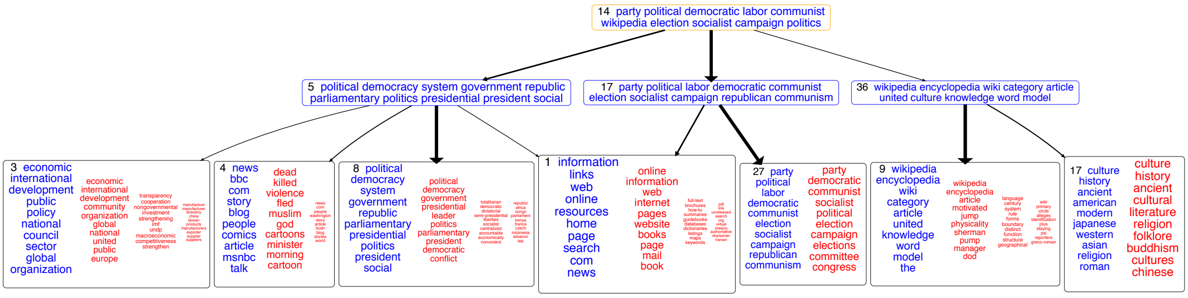

14 party political democratic labor communist wikipedia election socialist campaign politics
5 political democracy system government republic parliamentary politics presidential president social
17 party political labor democratic communist election socialist campaign republican communism
36 wikipedia encyclopedia wiki category article united culture knowledge word model
3 economic international development public policy national council sector global organization
4 news bbc com story blog people comics article msnbc talk
8 political democracy system government republic parliamentary presidential politics president social
1 information links web online resources home page search com news
27 party political labor democratic communist election socialist campaign republican communism
9 wikipedia encyclopedia wiki category article united knowledge word model the
17 culture history ancient american modern japanese western asian religion roman

Figure 3. Analogous plots to Figure 6 in the main paper for a tree about “movie” on the WS dataset.

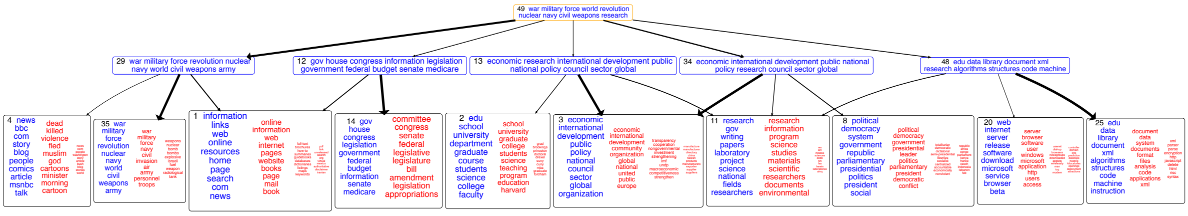

49 war military force world revolution nuclear navy civil weapons research
29 war military force revolution nuclear navy world civil weapons army
12 gov house congress information legislation government federal budget senate medicare
13 economic research international development public national policy council sector global
34 economic international development public national policy research council sector global
48 edu data library document xml research algorithms structures code machine
4 news bbc com story blog people comics article msnbc talk
35 war military force revolution nuclear navy world civil weapons army
1 information links web online resources home page search com news
14 gov house congress legislation government federal budget information senate medicare
2 edu school university department graduate course students science college faculty
3 economic international development public policy national council sector global organization
11 research gov writing papers laboratory project science national fields researchers
8 political democracy system government republic parliamentary presidential politics president social
20 web internet server release software download microsoft service browser beta
25 edu data library document xml algorithms structures code machine instruction

Figure 4. Analogous plots to Figure 6 in the main paper for a tree about “war” on the WS dataset.

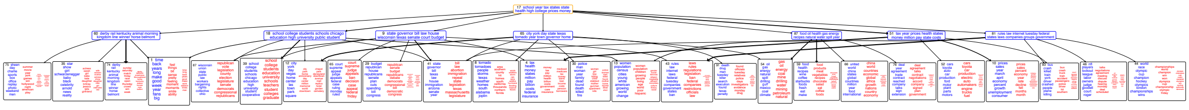


Figure 5. Analogous plots to Figure 6 in the main paper for a tree about “daily life” on the TMN dataset.

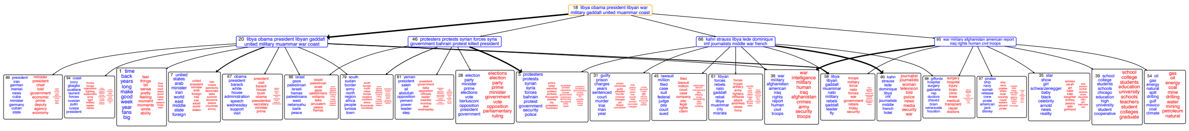


Figure 6. Analogous plots to Figure 6 in the main paper for a tree about “politics” on the TMN dataset.

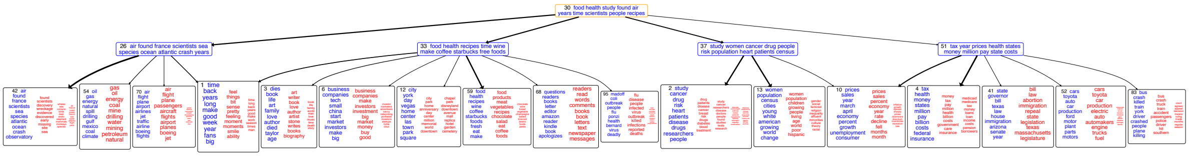


Figure 7. Analogous plots to Figure 6 in the main paper for a tree about “food and health” on the TMN dataset.

WEDTM

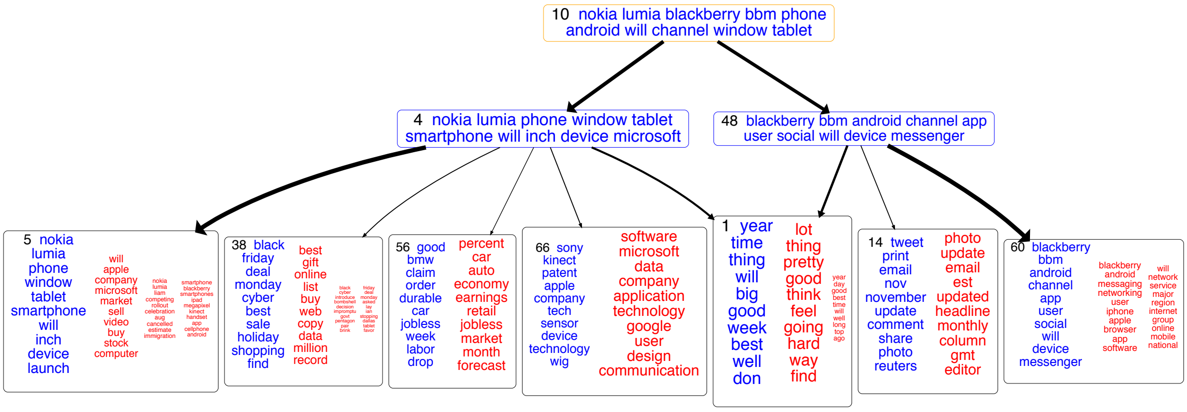

10 nokia lumia blackberry bbm phone android will channel window tablet
4 nokia lumia phone window tablet smartphone will inch device microsoft
48 blackberry bbm android channel app user social will device messenger
5 nokia lumia phone window tablet smartphone will inch device launch
will apple company microsoft market sell video buy stock computer
38 black friday deal monday cyber best sale holiday shopping find
best gift online list buy web copy data million record
56 good bmw claim order durable car jobless week labor drop
percent car auto economy earnings retail jobless market month forecast
66 sony kinect patent apple company tech sensor device technology wig
software microsoft data company application technology google user design communication
1 year time thing will big good week best well don
lot thing pretty good think feel going hard way find
14 tweet print email nov november update comment share photo reuters
photo update email est updated headline monthly column gmt editor
60 blackberry bbm android channel app user social will device messenger
blackberry android messaging networking user iphone apple browser app software
will network service major region internet group online mobile national

Figure 8. Analogous plots to Figure 6 in the main paper for a tree about “phone” on the Twitter dataset.

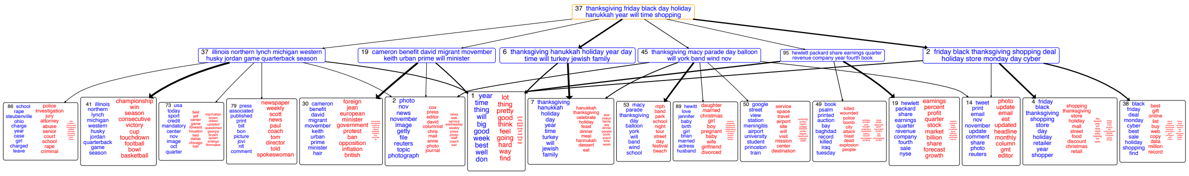

37 thanksgiving friday black day holiday hanukkah year will time shopping
37 illinois northern lynch michigan western husky jordan game quarterback season
19 cameron benefit david migrant movember keith urban prime will minister
6 thanksgiving hanukkah holiday year day time will turkey jewish family
45 thanksgiving macy parade day balloon will york band wind nov
95 hewlett packard share earnings quarter revenue company year fourth book
2 friday black thanksgiving shopping deal holiday store monday day cyber

Figure 9. Analogous plots to Figure 6 in the main paper for a tree about “thanksgiving” on the Twitter dataset.

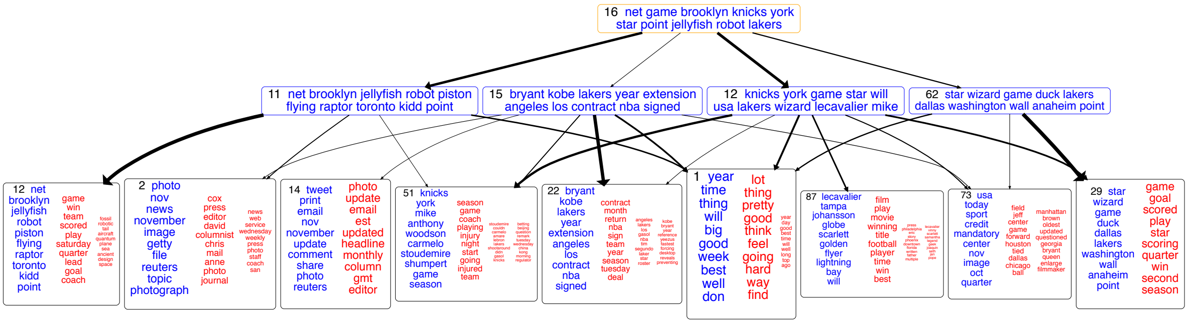

16 net game brooklyn knicks york star point jellyfish robot lakers
11 net brooklyn jellyfish robot piston flying raptor toronto kidd point
15 bryant kobe lakers year extension angeles los contract nba signed
12 knicks york game star will usa lakers wizard lecavalier mike
62 star wizard game duck lakers dallas washington wall anaheim point
12 net brooklyn jellyfish robot piston flying raptor toronto kidd point
game win team scored play saturday quarter lead goal coach
2 photo nov news november image getty file reuters topic photograph
cox press editor david columnist chris mail anne photo journal
14 tweet print email nov november update comment share photo reuters
photo update email est updated headline monthly column gmt editor
51 knicks york mike anthony woodson carmelo stoudemire shumpert game season
season game coach playing injury night start going injured team
22 bryant kobe lakers year extension angeles los contract nba signed
contract month return nba sign team year season tuesday deal
1 year time thing will big good week best well don
lot thing pretty good think feel going hard way find
87 lecavalier tampa johansson globe scarlett golden flyer lightning bay will
film play movie winning title football player time win best
73 usa today sport credit mandatory center nov image oct quarter
29 star wizard game duck dallas lakers washington wall anaheim point
game goal scored play star scoring quarter win second season

Figure 10. Analogous plots to Figure 6 in the main paper for a tree about “NBA” on the Twitter dataset.

lion microsoft market amazon tablet playstation maker devices wireless app technology security service sales users year smartphone android customers phones deal plans nokia report world week system online breach smartphones services amp apps group percent buy executive top thursday music information ceo computers samsung consumer location chief wednesday stores largest research business electronics growth books prices jobs china windows major selling hackers firm launch high digital tuesday friday shares intel companies systems tablets make years device city people population energy verizon blackberry profit internet bid store

Below are some example synthetic documents with WEDTM trained on the WS dataset.

1. research papers thesis project writing edu paper dissertation proposal reports patent technical report topics guide academic exploratory information write methods researchers publications parallel process phd issues ideas computing media topic custom projects essay help archives tools page labs web lab university communication resources ibm proposals intellectual home advice cluster dissertations survey library essays address practical links step abstract property bell systems linux performance master serve abstracts developed written department index quality exploring matters res theses documents course college collaborative processing programme materials focused preparing computer program students search guidelines online articles experience services recommendations welcome conduct funding aspects idea people

2. football team league news american fans nfl players soccer stadium sports indoor national com home statistics club history hockey baseball college fan association teams arena england game player texas website scores basketball professional fixtures espn season ncaa clubs university afl schedule tickets minnesota independent tables athletics sport welcome united leagues dedicated ice chicago giants nba stadiums online rugby stats nhl gymnastics coverage coaches levels delivers city arsenal index coach database conference standings information cstv fantasy bears articles views athletic moved rules assault bulls supporters officials kickoff division games women tournament photos body media fame covering adjacent sexual federation pages rutgers

3. health hiv prevention stress healthy aids diet nutrition hepatitis fitness food epidemic pressure infection information cdc disease people blood life gov diseases epidemiology treatment cholesterol picnic tips medical articles children safety symptoms fat topics heart foods living centers fatty help kids liver care researchers eating diagnosis who weight body virus recipes human control viral sheets com helps patterns and comprehensive diabetes risk trials int learn conditions advice influenza diets exercise affect public press consensus constant united eat infections listing unaids experts mind personality disorder causes world flu infectious guidelines global loss women avian humans population ucsf cope job brains index

4. system government parliamentary republic presidential maradona diego parliament systems westminster freedom independence democracy semi-presidential consensus constitutional armando elected congressional constitution executive america people declaration elections president czech gov australia election documents hand french political national congress authority united archives legislature kingdom country governments powers power public british representation canada argentine versus buenos separation born legislative central assembly affairs republics india peace france sri semi english branch body operates charters romania distributed presidency practice the parliaments governing house countries britain rule federal principles jury lanka foreign bill govern nunavut portuguese ireland aires immense head qld parl pub shtml politics ukraine parties

5. school research college edu graduate university education students elementary center student library district information schools program programs degree city public media papers phd master thesis law department undergraduate home project staff america regional writing scholarship calendar independent academic town paper administration news scholarships north dissertation community faculty welcome archives virginia children president union wisconsin temple proposal parents ohio pennsylvania teachers philadelphia reports guide houston nation degrees chronicle services massachusetts patent opportunities boston technical activities county study masters east report employment commission usa expo lewis michigan san academy memorial topics meeting institute campus homepage web arizona maine business announcements philly hub

References

Polson, N., Scott, J., and Windle, J. Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American Statistical Association*, 108(504): 1339–1349, 2013.

Wray, B. and Marcus, H. A Bayesian view of the Poisson-Dirichlet process. *arXiv preprint arXiv:1007.0296v2 [math.ST]*, 2012.

Zhao, H., Rai, P., Du, L., and Buntine, W. Bayesian multi-label learning with sparse features and labels, and label co-occurrences. In *AISTATS*, pp. 1943–1951, 2018.

Zhou, M. Softplus regressions and convex polytopes. *arXiv preprint arXiv:1608.06383*, 2016.

Zhou, M. and Carin, L. Negative binomial process count and mixture modeling. *TPAMI*, 37(2):307–320, 2015.

Zhou, M., Cong, Y., and Chen, B. Augmentable gamma belief networks. *JMLR*, 17(163):1–44, 2016.

Chapter 7

Sparse Bayesian Latent Factor Models for Multi-label Learning

A multi-label learning problem is an challenging supervised task which has many important applications such as image/document tagging, recommender system, and advertisements. As discussed in Section 3.3, a multi-label learning problem can be solved by BLFMs that factorise the label matrix conditioned on the feature matrix. In this chapter, I will introduce the proposed BLFM on multi-label learning with the binary feature matrix Zhao et al. [2018d]. Although the dimensions of the label matrix can be extremely large, most of the samples only have a tiny subset of the labels being active, meaning that the label matrix is usually very sparse. When it comes to binary feature, the feature matrix can also be very sparse. Therefore, by leveraging the two kinds of the sparsity of the feature and label matrices to facilitate fast inference, the proposed model can deal with the multi-label learning problem efficiently. In addition, the proposed model uses the label-label co-occurrence matrix to improve prediction performance when there are a significant fraction of missing labels.

The framework of the above model is shown in Figure 7.1, which can be viewed as the extension of the basic framework of BLFM shown in Figure 2.1 in Section 2.2.6 of Chapter 2. Specifically, the model factorises the label matrix into two latent matrices: the factor loading matrix and factor score matrix, where the former and latter are informed by the label-label correlations and the sample features, respectively.

The details of this research are shown in the following paper:

- **H. Zhao**, P. Rai, L. Du, W. Buntine, "Bayesian Multi-label Learning with Sparse Features and Labels, and Label Co-occurrences", in *Artificial Intelligence and Statistics* (**AISTATS**) 2018.

In terms of future research, similar to BLFMs in other areas, one important direction is further improving training speed for large-scale datasets. In addition, we need to take the efficiency of the testing phase into account. Note that in the testing phase of a multi-label learning problem, a model can only rely on a sample's features to predict its labels, which requires the model to estimate the posterior prediction distribution directly conditioned the features. This can be non-trivial for many BLFMs. Therefore, how to efficiently obtain the posterior prediction distribution, is another possible direction of future research.

The code of this research in this chapter is released at `https://github.com/ethanhezhao/BMLS`.

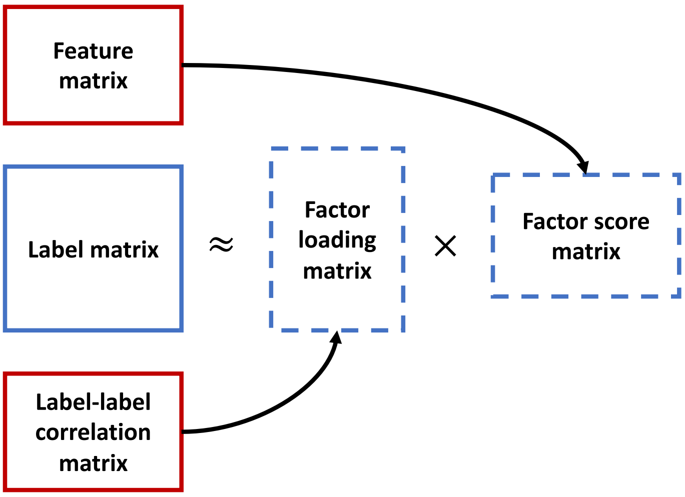


Figure 7.1: Model framework of Zhao et al. [2018d]. The blue rectangles with solid lines and dash lines are the data matrix (the label matrix) and the latent matrices, respectively. The red rectangles are the the feature matrix and the label-label correlation matrix, respectively.

Bayesian Multi-label Learning with Sparse Features and Labels, and Label Co-occurrences

He Zhao[*] **Piyush Rai**[†] **Lan Du**[*] **Wray Buntine**[*]

[*]Faculty of IT, Monash University, Australia
{he.zhao, lan.du, wray.buntine}@monash.edu

[†]Department of CSE, IIT Kanpur, India
piyush@cse.iitk.ac.in

Abstract

We present a probabilistic, fully Bayesian framework for multi-label learning. Our framework is based on the idea of learning a joint low-rank embedding of the label matrix and the label co-occurrence matrix. The proposed framework has the following appealing aspects: (1) It leverages the sparsity in the label matrix and the feature matrix, which results in very efficient inference, especially for sparse datasets, commonly encountered in multi-label learning problems, and (2) By effectively utilizing the label co-occurrence information, the model yields improved prediction accuracies, especially in the case where the amount of training data is low and/or the label matrix has a significant fraction of missing labels. Our framework enjoys full local conjugacy and admits a simple inference procedure via a scalable Gibbs sampler. We report experimental results on a number of benchmark datasets, on which it outperforms several state-of-the-art multi-label learning models.[1]

1 Introduction

Multi-label learning [Gibaja and Ventura, 2015, Prabhu and Varma, 2014, Jain et al., 2016, Babbar and Schölkopf, 2017] refers to the problem of learning to assign a subset of relevant labels to each object, given a large set of candidate labels. Each object is thus associated with a binary label vector, which denotes the presence/absence of each of the candidate labels. Multi-label learning problems are ubiquitous in a wide variety of applications, such as image/document tagging, recommender system, ad-placement.

In multi-label learning problems encountered in modern applications, it is common to have datasets characterized by instances defined by sparse, high-dimensional feature vectors, in addition to the corresponding label vectors themselves being sparse and high-dimensional. Moreover, often the label vector may be incomplete since it is usually not possible to completely annotate an instance with all of the relevant labels. Multi-label learning problems thus need to routinely deal with missing labels in the label vector of each training instance. Finally, scalability is another challenge in multi-label learning problems. Given the high degree of sparsity of features and labels, it is desirable to have multi-label learning algorithms that can leverage this sparsity during training/test time, and can consequently scale to large-scale problems.

Motivated by these issues and desiderata, we present a probabilistic framework for multi-label learning, which is capable of addressing these issues effectively, in a principled manner. Our framework is based on a generative latent factor model for the binary label matrix. This latent factor model is based on an efficient Poisson-Dirichlet-gamma non-negative factorization [Zhou et al., 2012] of the binary label matrix, which scales in the number of nonzeros in the label matrix. Moreover, we condition the latent factors on the instance features in a way that effectively utilizes the feature sparsity and further improves the scalability. Leveraging both instance label vector as well as instance feature vector sparsity leads to a very efficient inference for our model.

We further augment our model with a latent factor model for the label co-occurrences. Information about label co-occurrences can be obtained from an external source (e.g., a text corpus such as Wikipedia) and this information can be helpful, especially in predicting labels that are rare in the data (e.g., for which there are very training examples) or in cases where the label matrix could have a large fraction of labels as missing.

[1] Code at `https://github.com/ethanhezhao/BMLS`

Proceedings of the 21^{st} International Conference on Artificial Intelligence and Statistics (AISTATS) 2018, Lanzarote, Spain. PMLR: Volume 84. Copyright 2018 by the author(s).

Our latent factor model for the label co-occurrence is learned jointly with the latent factor model for the label matrix, and sharing the latent factors of the label helps in effectively transferring information from the label co-occurrences.

Our model enjoys local conjugacy, which leads to a very simple and highly efficient Bayesian inference via Gibbs sampling. Our model is considerably more scalable as compared to other state-of-the-art Bayesian models for multi-label learning, while achieving comparable and better prediction accuracies.

2 Background and Notation

In the multi-label learning problem, we assume that we are given an $D \times N$ instance feature matrix $\mathbf{X}$ and an $L \times N$ instance label matrix $\mathbf{Y} \in \{0,1\}^{L \times N}$, where N, D, L are the number of instances, the dimension of features, the dimension of labels, respectively. Both matrices are assumed to be highly sparse. In this paper, we focus on binary features, which are quite common, especially in large-scale multi-label learning tasks. An example would be in document classification: each instance is a text document which is associated with a binary feature vector indicating the presence/absence of words. The goal of multi-label learning is to use the feature matrix and the label matrix to learn a model that can predict the label vector $\boldsymbol{y}_*$, given the feature vector $\boldsymbol{x}_*$ of a new instance.

Our model is based on the idea of factorizing the label matrix $\mathbf{Y}$, which is equivalent to learning a low-dimensional embedding $\boldsymbol{\theta}_i$ for the label vector $\boldsymbol{y}_i$ (i.e., the i^{th} column vector of $\mathbf{Y}$) of each instance i [Yu et al., 2014, Rai et al., 2015, Mineiro and Karampatziakis, 2015]. The embedding $\boldsymbol{\theta}_i$ is, in turn, conditioned on the feature vector $\boldsymbol{x}_i$ (i.e., the i^{th} column vector of $\mathbf{X}$) associated with that instance. Given the feature vector of a new instance $\boldsymbol{x}_*$, its embedding $\boldsymbol{\theta}_*$ can be computed and it label vector $\boldsymbol{y}_*$ can be predicted/decoded from $\boldsymbol{\theta}_*$. Different label embedding models vary in how the embeddings are conditioned on the features and how the embeddings are decoded to produce the label vector at test time.

Our model has the following distinguishing aspects as compared to other existing label embedding methods for multi-label learning: (1) Learning the embeddings by our model scales in the number of nonzeros in the label and feature matrices, and (2) The model can effectively leverage the label co-occurrence matrix, if available. The latter property is especially useful when a significant fraction of the labels are missing in the label matrix and/or if the number of training instances are very small.

3 The Model

Our model assumes that each entry $y_{l,i} \in \{0,1\}$ of the label matrix $\mathbf{Y}$ is generated by first drawing a *latent* count $z_{l,i}$ from the Poisson distribution with rate parameter $\psi_{l,i}$ and then thresholding the count at 1.

$$\begin{aligned} y_{l,i} &= \mathbf{1}_{z_{l,i}>0} && (1) \\ z_{l,i} &\sim \text{Poisson}(\psi_{l,i}) && (2) \end{aligned}$$

where $\mathbf{1}.$ is the indicator function. To assist clarity, we further denote the latent count matrix as $\mathbf{Z} \in \mathbb{Z}^{L \times N}$ and the Poisson rate matrix as $\boldsymbol{\Psi} \in \mathbb{R}_+{}^{L \times N}$.

By integrating $z_{l,i}$ out, the above generative process for $y_{l,i}$ can be shown to be equivalent to

$$y_{l,i} \sim \text{Bernoulli}[1 - \exp(-\psi_{l,i})] \quad (3)$$

which is the Bernoulli-Poisson (BP) link function [Zhou, 2015] for binary observations. A particularly appealing aspect of the BP link (as opposed to other link function for binary observations, such as logistic/probit) is that the inference cost only depends on the number of nonzeros in the data [Zhou, 2015], making it an ideal choice for the problems involving the large-scale multi-label learning problems with sparsity. Specifically, if the $y_{i,l} = 0$, $z_{i,l} = 0$ with probability one. Therefore we only need to infer the latent count $z_{i,l}$ for those labels $y_{i,l}$ that are nonzero. That is how the sparsity of the label matrix is leveraged in our model.

3.1 A Low-Rank Model for Label Matrix

Most real-world multi-label learning datasets consist of high-dimensional labels vectors. However, the labels tend to be related to each other. Therefore, a popular assumption used in multi-label learning is to use a low-rank approximation for the label matrix, as also used in recent work [Yu et al., 2014, Rai et al., 2015, Mineiro and Karampatziakis, 2015, Bhatia et al., 2015]. To this end, we assume that the Poisson parameter matrix $\boldsymbol{\Psi}$ admits a low-rank factorization as follows:

$$\boldsymbol{\Psi} = \boldsymbol{\Phi}^\top \boldsymbol{\Theta} \quad (4)$$

where $\boldsymbol{\Theta} \in \mathbb{R}_+{}^{K \times N}$ and $\boldsymbol{\Phi} \in \mathbb{R}_+{}^{K \times L}$.

For one instance i, the model can be written as:

$$\begin{aligned} \boldsymbol{y}_i &\sim \text{Bernoulli}[1 - \exp(-\boldsymbol{\psi}_i)] && (5) \\ \boldsymbol{\psi}_i &= \boldsymbol{\Phi}^\top \boldsymbol{\theta}_i = \sum_{k=1}^{K} \boldsymbol{\phi}_k \theta_{i,k} && (6) \end{aligned}$$

The model can be interpreted as follows: The label vector $\boldsymbol{y}_i$ is associated with an embedding $\boldsymbol{\theta}_i$ and $\boldsymbol{\Phi}$

can be considered as K "topics", each a distribution over the L labels. The label vector $\boldsymbol{y}_i$ of instance i can then be thought of as being generated via a linear combination of these K topics through the BP link. The combination weights given by the embedding vector $\boldsymbol{\theta}_i$, with $\theta_{k,i}$ representing the weight of topic k, where $\phi_{k,l}$ represents the weight of label l in topic k. Finally, we impose Dirichlet prior on $\boldsymbol{\phi_k}$:

$$\boldsymbol{\phi_k} \sim \text{Dirichlet}_L(\beta_0, \cdots, \beta_0) \tag{7}$$

3.2 Conditioning Embeddings on Features

To condition the label vector embeddings $\boldsymbol{\theta}_i$ on the feature vector $\boldsymbol{x}_i$, we model $\theta_{k,i}$ a log-linear combination of the instance's features as follows:

$$\theta_{k,i} = b_k \prod_{d}^{D} h_{k,d}^{x_{d,i}} \tag{8}$$

where $h_{k,d} \in \mathbb{R}_+$ is a latent variable controlling the influence of feature d on topic k and $b_k \in \mathbb{R}_+$ is a feature-independent bias term. Both $h_{k,d}$ and b_k are drawn from a gamma distribution:

$$h_{k,d}, b_k \sim \text{Gamma}(\mu_0, 1/\mu_0) \tag{9}$$

Figure 1 shows the graphical model for the above construction. Given our model construction, $h_{k,d}$ is expected to have mean 1. The intuition is that, in multi-label learning problems, the number of features D is usually very large but, for most of the instances, only a small subset of these features is discriminative. Therefore, if feature d does not contribute to topic k or is not very informative, then $h_{k,d}$ should be dominated by the prior and expected to be near 1, in order to have little influence on $\theta_{k,i}$. Note that the variance of $h_{k,d}$ is $\frac{1}{\mu_0}$, which is a hyperparameter of our model.

One of the particularly appealing aspects of our parameterization in Eq. 8 is its computational efficiency when the features are sparse (which is usually the case with most multi-label learning datasets). In contrast, the existing label embedding models [Yu et al., 2014, Rai et al., 2015, Mineiro and Karampatziakis, 2015] learn an explicit regression model from the D dimensional feature vector $\boldsymbol{x}_i$ to $\theta_{i,k}$, which is computationally very expensive for large D. At the same time, the choice of parameterization in Eq. 8 also facilitates in retaining the conjugacy of our model, leading to a simple and efficient inference algorithm. We will study the details of how the inference leverages the sparsity of the feature matrix in Section 4.

3.3 Leveraging Label Co-occurrences

In addition to the labels of the instances, it is often possible to get *label co-occurrence* statistics [Mensink et al., 2014] from an external source, such as a text corpus (e.g., Wikipedia). Suppose the label co-occurrence statistics are provided in form of an $L \times L$ count matrix $\mathbf{C} \in \mathbb{Z}^{L\times L}$, where each entry of $\mathbf{C}$ denotes the number of times a pair of labels co-occurs. Note that in the absence of an external source of information, one possible way to construct the matrix $\mathbf{C}$ could be to use the label matrix $\mathbf{Y}$ itself, i.e., as $\mathbf{C} = \mathbf{Y}^\top\mathbf{Y}$. In this case, even though $\mathbf{C}$ reuses the information already present in $\mathbf{Y}$, this "re-encoding" of information can still help the model, as also corroborated by recent work [Liang et al., 2016].

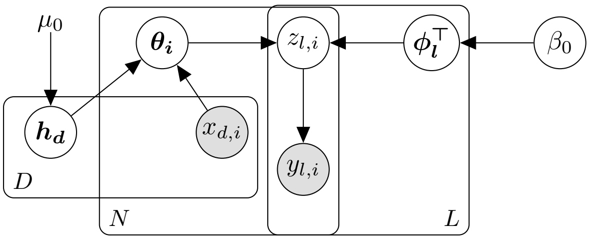


Figure 1: The graphical model for factorizing the label matrix. $\boldsymbol{h_d}$, $\boldsymbol{\theta_i}$, $\boldsymbol{\phi_l}$ is the d^{th} column of $\mathbf{H}$, the i^{th} column of $\boldsymbol{\Theta}$, the l^{th} row of $\boldsymbol{\Phi}$ respectively. All of them are K dimensional vectors.

It is natural to model label co-occurrences by the Poisson distribution:

$$c_{l,m} \sim \text{Poisson}(\psi'_{l,m}) \tag{10}$$

where $c_{l,m}$ denotes the number of times a pair of labels l and m co-occurs, $\psi'_{l,m}$ denotes the $(l,m)^{th}$ entry in the Poisson rate matrix $\boldsymbol{\Psi'} \in {\mathbb{R}_+}^{L\times L}$. We further apply a low-rank factorization of $\boldsymbol{\Psi'}$ as follows:

$$\boldsymbol{\Psi'} = \boldsymbol{\Phi}^\top\boldsymbol{\Lambda}\boldsymbol{\Phi} \tag{11}$$

Here $\boldsymbol{\Lambda} \in {\mathbb{R}_+}^{K\times K}$ is a diagonal matrix, whose diagonal elements are denoted by the vector $\boldsymbol{\lambda} \in {\mathbb{R}_+}^{K}$. We assume λ_k to have a gamma prior distribution:

$$\lambda_k \sim \text{Gamma}(\gamma_0/K, f_0) \tag{12}$$

where γ_0, f_0 are given uninformative gamma priors.

Figure 2 shows the graphical model of this part. Note that $\boldsymbol{\Phi}$ in Eq. (11) is the same "K topics" matrix that we have used in the low-rank modeling of the label matrix $\mathbf{Y}$ (Sec. 3.1). This is essentially a co-factorization model, such as the *collective matrix factorization* Singh and Gordon [2010], Klami et al. [2013], for joint low-rank modeling of multiple matrices with shared latent factors. In our case, these matrices are the label matrix $\mathbf{Y}$ and the label co-occurrence matrix $\mathbf{C}$, with the topic matrix $\boldsymbol{\Phi}$ shared by the latent factor models of both $\mathbf{Y}$ and $\mathbf{C}$. Note however that

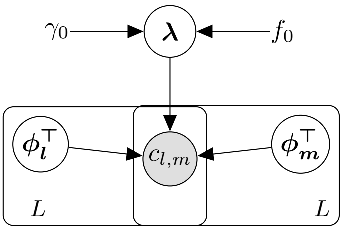


Figure 2: The graphical model for leveraging co-occurrences. $\boldsymbol{\phi_l}$ is the l^{th} row of $\boldsymbol{\Phi}$. $\boldsymbol{\phi_l}$ and $\boldsymbol{\lambda}$ are K dimensional vectors. Note: In the overall model, this part is learned jointly with the factorization of the label matrix.

unlike collective matrix factorization Singh and Gordon [2010], Klami et al. [2013], our gamma-Poisson generative model can effectively leverage the sparsity of these matrices and results in very efficient inference, with complexity that scales in the number of nonzeros.

4 Inference

Exact inference in our Bayesian model is intractable. However, one of the most appealing properties of our model is that it admits very simple yet efficient approximate inference via closed form Gibbs sampling updates. Leveraging data augmentation techniques Zhou et al. [2012], the proposed model enjoys full local conjugacy and facilitates deriving efficient Gibbs sampling updates for all the latent variables of our model. Moreover, the inference in our model scales in the number of nonzeros in both the label matrix as well as the feature matrix, which makes the model work efficiently for multi-label learning problems that involve large but highly sparse feature and label matrices.

4.1 Sampling Latent Counts $z_{l,i,k}$

Given a binary label $y_{l,i}$, according to our model construction in Eq. (2), we first need to sample the corresponding latent count $z_{l,i}$, which can be drawn from a truncated Poisson distribution:

$$(z_{l,i}|y_{l,i}, \psi_{l,i}) \sim y_{l,i} \cdot \text{Poisson}_{+}(\psi_{l,i}) \tag{13}$$

The above equation indicates that we only need to sample $z_{l,i}$ if $y_{l,i} > 0$, i.e., the sparsity of the label matrix.

Given Eq. (2) and the additivity of Poisson, the latent count $z_{l,i}$ can be written as a sum of K smaller latent counts, each of which is contributed by the corresponding topic:

$$z_{l,i} = \sum_{k}^{K} z_{l,i,k} \tag{14}$$

$$z_{l,i,k} \sim \text{Poisson}(\phi_{k,l}\theta_{k,i}) \tag{15}$$

where $z_{l,i,k}$ is the counts for each topic k.

Moreover, using the relationship of the Poisson and multinomial distributions, we can express the decomposition in Eq. (14) and Eq. (15) as a draw from a multinomial:

$$[z_{l,i,1}, \cdots, z_{l,i,K}] \sim \text{Multi}\left\{z_{l,i}; \frac{[\phi_{1,l}\theta_{1,i}, \cdots, \phi_{K,l}\theta_{K,i}]}{\sum_{k}^{K} \phi_{k,l}\theta_{k,i}}\right\} \tag{16}$$

4.2 Sampling Latent Counts $c_{l,m,k}$

To infer the latent factors defining the generative model of the count-valued label co-occurrences $c_{l,m}$ (Fig. 2), we leverage a similar latent variable augmentation scheme to the one used for sampling the latent counts associated with the label matrix (cf., Section 4.1). In particular, we assume the *observed* label co-occurrence $c_{l,m}$ for two labels l and m as a sum of K smaller latent counts (each of which can be attributed to one of these K topics) as follows

$$c_{l,m} = \sum_{k}^{K} c_{l,m,k} \tag{17}$$

$$c_{l,m,k} \sim \text{Poisson}(\phi_{k,l}\lambda_k\phi_{k,m}) \tag{18}$$

where $c_{l,m,k}$ is the latent counts for topic k.

Again, given $c_{l,m}$, which is observed, $c_{l,m,k}$ can be sampled from multinomial, similar to the sampling of $z_{l,i,k}$ in Eq. (16).

4.3 Sampling $h_{k,d}$ and b_k

As $\boldsymbol{\phi_k}$ is normalized (sums to 1), summing over l of Eq. (15) and using the additivity of Poisson, we get:

$$z_{\cdot,i,k} \sim \text{Poisson}(\theta_{k,i}) \tag{19}$$

where $z_{\cdot,i,k} = \sum_{l}^{L} z_{l,i,k}$. Thus, the likelihood of θ is

$$\prod_{k,i} e^{-\theta_{k,i}}\theta_{k,i}^{z_{\cdot,i,k}} \tag{20}$$

Given Eq. (8), recall that all the features are binary and $h_{k,d}$ influences $\theta_{k,i}$ iff $x_{d,i} = 1$. This gives us a direct way of extracting $h_{k,d}$ from $\theta_{k,i}$. We can derive the likelihood of $h_{k,d}$ as:

$$e^{-h_{k,d}\sum_{i:x_{d,i}=1}^{N}\frac{\theta_{k,i}}{h_{k,d}}}(h_{k,d})^{\sum_{i}^{N} x_{d,i}z_{\cdot,i,k}} \tag{21}$$

which is conjugate to its Gamma prior. Therefore, it is straightforward to yield the following sampling strategy for $h_{k,d}$:

$$h_{k,d} \sim \text{Gamma}\left(\mu_0 + \sum_{i:x_{d,i}=1}^{N} z_{\cdot,i,k}, \frac{1}{\mu_0 + \sum_{i:x_{d,i}=1}^{N} \frac{g_{k,i}}{h_{k,d}}}\right) \tag{22}$$

b_k can be sampled using the same formula by adding an extra row of ones in the feature matrix $\mathbf{X}$ (which serve as the default features).

We can compute and cache the value of $\theta_{k,i}$ first. After $h_{k,d}$ is sampled, we can update $\theta_{k,i}$ for the instances where feature d is on:

$$\theta_{k,i} \leftarrow \frac{\theta_{k,i} h'_{k,d}}{h_{k,d}} \tag{23}$$

where $h'_{k,d}$ is the newly-sampled value of $h_{k,d}$.

To sample h and compute θ, according to Eq. (8) and Eq. (22), one only iterates over the instances where feature d is on (i.e., $x_{d,i} = 1$) instead of iterating over all the instances. This demonstrates how the sparsity in the feature matrix is leveraged. Note that the inference simplicity only exists with binary features.

4.4 Sampling $\boldsymbol{\phi_k}$

If the co-occurrence matrix is not incorporated, using Eq. (16) and the Dirichlet-multinomial conjugacy, $\boldsymbol{\phi_k}$ can be sampled as:

$$\boldsymbol{\phi_k} \sim \text{Dirichlet}_L(\beta_0 + z_{1,\cdot,k}, \cdots, \beta_0 + z_{L,\cdot,k}) \tag{24}$$

where $z_{l,\cdot,k} = \sum_i^N z_{l,i,k}$.

Otherwise, ϕ is also involved in the generative process of $\mathbf{C}$. According to Eq. (18), the likelihood of $\mathbf{C}$ is

$$e^{-\sum_{l,m,k} -\phi_{k,l}\lambda_k\phi_{k,m}} \prod_{l,m,k} (\phi_{k,l}\lambda_k\phi_{k,m})^{c_{l,m,k}} \tag{25}$$

Given the fact that $\boldsymbol{\phi_k}$ is normalized, the likelihood term related to $\phi_{k,l}$ is: $\phi_{l,k}^{c_{l,\cdot,k}}$ where $c_{l,\cdot,k} = \sum_m^L c_{l,m,k} + \sum_m^L c_{m,l,k}$. Therefore, we can sample $\boldsymbol{\phi_k}$ as:

$$\boldsymbol{\phi_k} \sim \text{Dirichlet}_L(\cdots, \beta_0 + z_{l,\cdot,k} + c_{l,\cdot,k}, \cdots) \tag{26}$$

4.5 Sampling λ_k

According to Eq. (25), λ_k has the Poisson likelihood, which is conjugate to its Gamma prior. Therefore, we can sample λ_k as:

$$\lambda_k \sim \text{Gamma}[\gamma_0/K + c_{\cdot,\cdot,k}, 1/(f_0 + 1)] \tag{27}$$

where $c_{\cdot,\cdot,k} = \sum_l^L c_{l,\cdot,k}$.

Recall that γ_0 and f_0 have uninformative Gamma prior. For γ_0, we can apply the data augmentation in Zhou et al. [2012], Buntine and Hutter [2012] to get the Gamma likelihood. For f_0, its posterior is directly conjugate to the Gamma likelihood.

4.6 Time-Complexity Analysis

In addition to having a rich generative model for the label and label co-occurrences, one of the key properties of the proposed model is the computational efficiency resulting from taking advantage of the sparsity in both feature and label matrices. This is important because in many multi-label learning problems, the feature and label matrices usually are massive but highly sparse. Specifically, for the label matrix, with the Bernoulli-Poisson link, the models scales in the number of nonzeros in the label matrix. At the same time, sampling h and computing θ scale in the number of nonzeros in the feature matrix. Therefore, in the case where the label co-occurrences are not leveraged, the inference complexity of the proposed model is $\mathcal{O}(KG + KDG')$ where G is the number of nonzeros in the label matrix $\mathbf{Y}$ and G' is the average number of instances where a feature is on (i.e., the column-wise sparsity of $\mathbf{X}$). Even when the label co-occurrences are leveraged, it does not add much overhead since the label co-occurrence matrix is usually highly sparse as well and its low-rank factorization scales in the number of nonzeros in this matrix. The efficiency of our model will be empirically studied in Section 6.4.

5 Related Work

Multi-label learning problems in modern-day applications are usually characterized by a large number of training instances, a large number of features, and a large number of labels (i.e., label-space cardinality). Owing to this, there is a considerable recent interest in designing multi-label learning models that can gracefully scale to handle such large datasets.

Label embedding methods offer an appealing solution to the large label-space cardinality problem. These methods project the high-dimensional sparse label vectors of each instance into a low-dimensional space. This corresponds to learning a low-rank embedding of the label matrix. However, learning the embedding itself is a computationally challenging problem, especially when the label matrix is massive. This has led to a lot of recent interest in embedding based models for multi-label learning that can learn label matrix embeddings efficiently [Yu et al., 2014, Mineiro and Karampatziakis, 2015]. However, most of these methods do not exploit the sparsity of the label matrix while learning the embeddings. Recently, [Rai et al., 2015] proposed a Bayesian label matrix embedding method that scales in the number of nonzeros in the label matrix. Their approach is similar in spirit to our approach. However, the approach in [Rai et al., 2015] conditions the embeddings on the feature vectors via a regression model. Learning this regression model is challenging due to non-conjugacy, and is computationally expensive. In contrast, our approach of learning the label matrix embedding also scales in the number of nonzeros in the label matrix. However, the embeddings are conditioned on the feature vector not via a regression model used in [Rai et al., 2015] but via a log-linear combination of the features. If the features are binary and sparse, such an approach of conditioning on the features leads to significant speed-ups. In our experiments, we compare the per iteration computational cost of our approach with the approach of [Rai et al., 2015] and observe significant speed-ups. Moreover, unlike our model, the model of [Rai et al., 2015] cannot leverage label co-occurrences.

Other prominent Bayesian approaches to multi-label include the Bayesian compressed sensing (BCS) based approach [Kapoor et al., 2012]. However, inference in BCS is expensive. Moreover, it does not exploit the sparsity of label matrix or feature matrix, and is therefore not suitable for large-scale multi-label datasets.

Leveraging label co-occurrences to improve multi-label learning has not received much attention so far, except for some recent works such as [Mensink et al., 2014, Gaure et al., 2017]. One key difference of our model as compared to these models is that the computational cost scales in the number of nonzeros in the label and feature matrix. Moreover, the Poisson-Dirichlet-gamma based latent factor model offers a nice interpretability of our model, making it also suitable for other tasks, such as topic discovery (e.g., group of related labels representing a topic). In our experiments, we show such a qualitative analysis on a real dataset.

Our approach of constructing embeddings via conditioning on features is related to the models that incorporate auxiliary information in Poisson factorization or topic models such as the ones in Hu et al. [2016], Zhao et al. [2017a,b,c]. Features in those models are used to construct the prior of the embeddings. However, in our model, the embeddings are directly constructed using the features (Eq. 8), which allows efficiently computing the embeddings of test instances.

6 Experiments

In our experiments, we compare the proposed **B**ayesian **M**ulti-label **L**earning with **S**parse Features and Labels (abbreviated BMLS) with various state-of-the-art multi-label learning models, which include both Bayesian and non-Bayesian models. We evaluate the proposed model on four benchmark multi-label datasets with binary features: Bibtex, Delicious, Movielens, and NIPS.

The statistics of the datasets are listed in Table 1. The datasets cover a wide range of feature and label sizes. Moreover, both the feature vectors as well as the label vectors are highly sparse, reflecting real-world multi-label learning problems. Our model can effectively exploit the sparsity in these vectors, which results in a fast inference procedure.

We compare the following models: **(1) BMLS:** Our proposed model. We experiment with two variants - with and without the label co-occurrences. If the label co-occurrences are leveraged, we refer to the model as BMLS-co. **(2) LEML:** Low rank Empirical risk minimization for Multi-label Learning Yu et al. [2014]. Similar to our model, LEML factorizes the label matrix $\mathbf{Y}$ with two matrices and one of them is further factorized with the feature matrix $\mathbf{X}$. LEML considers various types of loss functions such as squared loss, logistic loss, hinge loss, etc. **(3) BMLPL:** Bayesian Multi-label Learning via Positive Labels Rai et al. [2015]. As one of the most related models to BMLS, BMLPL applies the Bernoulli-Poisson factorization on $\mathbf{Y}$ as well. However, unlike our model, BMLPL uses a regression based approach to condition on the features. **(4) BCS:** Bayesian Compressed Sensing for multi-label learning Kapoor et al. [2012]. BCS is a Bayesian method that uses the idea of doing compressed sensing on the label vectors Hsu et al. [2009], and relies on variational inference. **(5) BNMC:** Bayesian Nonparametric model for Multi-label Classification Nguyen et al. [2016]. BNMC is a Bayesian model that automatically learns and exploit the unknown number of multi-label correlation.

We report the Area Under the ROC Curve (AUC) on the test data to measure the prediction performance on new instances for all the models being compared. In particular, for our model, we can obtain $\mathbf{H}, \boldsymbol{b}, \mathbf{\Phi}$ from the training phase. Given a new instance i', we can compute $\theta_{k,i'}$ by Eq. (8) using its feature vector $\boldsymbol{x}_{i'}$. The labels can be predicted as follows:

$$\Pr(y_{l,i'} = 1) = 1 - e^{-\sum_k^K \phi_{k,l}\theta_{k,i'}}$$

In the experiments, we set the hyperparameters for our model as $\mu_0 = 10$, $\beta_0 = 0.01$, $K = 100$ and γ_0, f_0 are given uninformative gamma priors. We use 5000 Gibbs sampling iterations to train the model and report the average results over the last 2500 iterations. For the baseline models, we use their default parameter settings.

Table 1: The statistics of the datasets used in the experiments. N_{train}: number of training instances, N_{test}: number of test instances, D: number of features, L: number of labels.

Dataset	N_{train}	N_{test}	D	L
Bibtex	4880	2515	1836	159
Delicious	12920	3185	500	983
Movielens	4000	2040	29	3952
NIPS	2292	573	2484	14036

Table 2: Comparison of the various methods in terms of AUC scores with all the instances in the training sets. "-" denotes either these results were not available or the method was infeasible to run on that data set.

Model	Bibtex	Delicious	Movielens	NIPS
LEML	0.9040	0.8894	**0.8787**	0.8777
BMLPL	0.9210	0.8950	0.8582	0.9002
BCS	0.8614	0.8000	-	-
BNMC	0.8318	-	-	-
BMLS	**0.9379**	**0.9062**	0.8682	**0.9009**

6.1 Results using Complete Training Set

In the first experiment, we train all the models using all the instances in the training set. The AUC scores are reported in Table 2. The result shows that the proposed model performs better than the other models in three out of four datasets, which evidences the effectiveness of our model. Note that BMLS-co performs comparably to BMLS in this setting (possibly because training data is plenty), so its results are not reported.

6.2 Results using Missing Labels and Limited Training Instances

One common problem of multi-label learning is missing labels. As a Bayesian model, the proposed model naturally handles this problem. Furthermore, it is reasonable to assume that the label co-occurrences shall play a more important role in the case of missing labels. To examine this, we randomly remove 80% entries from the label matrix in the training data of Bibtex, Delicious, and Movielens to mimic the situation where a significantly large fraction of the labels are missing. The AUC scores of this experiment are shown in Table 3. From the results, it can be observed that BMLS-co gains better results than BMLS, especially on the Bibtex dataset, demonstrating that the label co-occurrences do help in the case with missing labels. Moreover, both of our proposed models outperform the others significantly in this case. It is also noteworthy that although LEML gets better AUC score on the Movielens dataset with all the training instances, the proposed models have a clear advantage when there is a high fraction of missing labels.

Table 3: AUC scores with only 20% labels.

Model	Bibtex	Delicious	Movielens
LEML	0.8452	-	0.8406
BMLPL	0.7879	0.8082	0.8574
BMLS	0.8598	0.8933	0.8619
BMLS-co	**0.8764**	**0.8978**	**0.8643**

Table 4: AUC scores with only 20% instances of the training set.

Model	Bibtex	Delicious	Movielens
LEML	0.8649	0.7325	0.8429
BMLPL	0.8167	0.8484	0.8437
BNMC	0.7549	-	-
BMLS	0.8651	0.8888	**0.8629**
BMLS-co	**0.8723**	**0.8921**	0.8562

Another situation where the label co-occurrences may benefit is the case where there are not sufficient training examples in the data. We mimic this situation by reducing the size of training instances to 20% on Bibtex, Delicious, and Movielens. The AUC scores in this case in shown in Table 4. Here we can observe a similar trend as for the missing label case: BMLS has significantly better performance as compared to the baseline models and BMLS-co further improves the prediction accuracies using the label co-occurrences.

6.3 Qualitative Analysis: Topic Modeling on NIPS Dataset

Recall that in our model, $\boldsymbol{\phi}_k$ represents a distribution (i.e., a "topic") over the labels. To assess our model's ability to discover meaningful topics, we run an experiment on the NIPS dataset with $K = 100$ and examine each topic. The NIPS dataset consists of 14036 labels (each of which is a word; each author (i.e., instance) has a subset of words), so $\boldsymbol{\phi}_k$ is of that size. In Table 5, we show five of the topics with their top words (ranked by $\phi_{k,l}$) and the top authors (ranked by $\theta_{k,i}$). As shown in the table, our model is able to discover clear and meaningful topics of the authors, which shows its usefulness as a topic model when each document $\boldsymbol{y}_i \in \{0,1\}^L$ has features in form of meta data $\boldsymbol{x}_i \in \{0,1\}^D$ associated with it.

Table 5: The top words and authors with the largest weights in the topics.

Topic:	1	2	3	4	5
Top words:	input neural networks network training set learning output weights information	problem theorem theory bound result exists positive dimension proof assume	image dimensional system vision images visual object computer pattern position	posterior distributions log likelihood monte inference bayesian joint carlo variance	optimal control current actions dynamic programming learn action state machine
Top authors:	Mozer_M Hinton_G Sejnowski_T Bengio_Y Giles_C	Sontag_E Venkatesh_S Bartlett_P Jordan_M Meir_R	Sejnowski_T Hinton_G Baluja_S Zemel_R Poggio_T	Jordan_M DeFreitas_J Hinton_G Doucet_A Bishop_C	Sejnowski_T Dayan_P Hinton_G Mozer_M Jordan_M

6.4 Running Time

In this section, we empirically compare the running time of our model with BMLPL[2], with a similar low-rank embedding approach. Note that BMLPL uses a regression approach to condition the embeddings on the features, while in our model, the embeddings are conditioned on the features via a log-linear combination of the features. This makes our model much more scalable, while also enjoying closed form, highly efficient Gibbs sampling.

Both the models are implemented in MATLAB running on a desktop with 3.40 GHz CPU and 16GB RAM. We report the running time per MCMC iteration on the four datasets and we also vary the size of training instances from 20% to 80% to fully exam the efficiency. Shown in Table 6, the proposed model runs much faster than BMLPL, supporting the time-complexity analysis in Section 4.6.

Table 6: Running time per iteration (seconds) of BMLS and BMLPL. $K = 100$ for both models.

Dataset	% training	BMLPL	BMLS
Bibtex	20%	18.14	0.04
	40%	22.54	0.06
	60%	26.75	0.09
	80%	29.80	0.11
Delicious	20%	12.18	0.09
	40%	14.45	0.16
	60%	17.82	0.24
	80%	20.70	0.33
Movielens	20%	19.19	0.16
	40%	21.86	0.27
	60%	24.08	0.37
	80%	26.27	0.49
NIPS	20%	35.50	0.66
	40%	38.51	1.10
	60%	40.31	1.55
	80%	43.06	2.01

7 Conclusion and Discussion

Despite the considerable amount of recent progress on the problem of multi-label learning, Bayesian approaches to this problem have received relatively little attention. This is primarily due to the lack of scalable approaches that can handle large datasets and can be efficient at training and test time. With this motivation, in this paper, we presented a framework for multi-label learning that leverages some of the key characteristics of multi-label learning datasets (in particular, the sparsity of label and feature matrix) to design a scalable Bayesian multi-label learning model. Unlike most existing multi-label learning models that are based on learning a low-rank factorization of the label matrix, our model performs a joint factorization of the label matrix and the label co-occurrence matrix and, by sharing latent factors between the two factorizations, it can address problems such as lack of training data and/or a high fraction of missing labels in the label matrix. The topic-based interpretation of our label embedding approach is intuitive and we hope it would motivate the application of similar topic model based approaches for the problem of multi-label learning. Finally, making such models more scalable would be an interesting direction of future work. Although in this paper, we have presented Gibbs sampling for doing inference in the model, developing variational inference or stochastic variational inference would further improve the scalability of our model.

Acknowledgements

PR acknowledges support from IBM Faculty Award, DST-SERB Early Career Research Award, and Dr. Deep Singh and Daljeet Kaur Faculty Fellowship, IIT Kanpur.

[2] We only compare the running time with BMLPL because (1) it is a Bayesian model with the similar base framework like ours, (2) its inference is done by Gibbs sampling and implemented in MATLAB as well.

References

R. Babbar and B. Schölkopf. DiSMEC-distributed sparse machines for extreme multi-label classification. In *WSDM*, 2017.

K. Bhatia, H. Jain, P. Kar, M. Varma, and P. Jain. Sparse local embeddings for extreme multi-label classification. In *NIPS*, 2015.

W. Buntine and M. Hutter. A Bayesian view of the Poisson-Dirichlet process. *arXiv preprint arXiv:1007.0296v2 [math.ST]*, 2012.

A. Gaure, A. Gupta, V. K. Verma, and P. Rai. A probabilistic framework for zero-shot multi-label learning. In *UAI*, 2017.

E. Gibaja and S. Ventura. A tutorial on multilabel learning. *ACM Comput. Surv.*, 2015.

D. J. Hsu, S. M. Kakade, J. Langford, and T. Zhang. Multilabel prediction via compressed sensing. In *NIPS*, 2009.

C. Hu, P. Rai, and L. Carin. Non-negative matrix factorization for discrete data with hierarchical side-information. In *19th International Conference on Artificial Intelligence and Statistics*, pages 1124–1132, 2016.

H. Jain, Y. Prabhu, and M. Varma. Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In *KDD*, 2016.

A. Kapoor, R. Viswanathan, and P. Jain. Multilabel classification using bayesian compressed sensing. In *NIPS*, 2012.

A. Klami, G. Bouchard, and A. Tripathi. Group-sparse embeddings in collective matrix factorization. *CoRR*, abs/1312.5921, 2013.

D. Liang, J. Altosaar, L. Charlin, and D. M. Blei. Factorization meets the item embedding: Regularizing matrix factorization with item co-occurrence. In *RecSys*, 2016.

T. Mensink, E. Gavves, and C. G. Snoek. Costa: Co-occurrence statistics for zero-shot classification. In *CVPR*, 2014.

P. Mineiro and N. Karampatziakis. Fast label embeddings via randomized linear algebra. In *ECML*, 2015.

V. Nguyen, S. Gupta, S. Rana, C. Li, and S. Venkatesh. A Bayesian nonparametric approach for multi-label classification. In *ACML*, 2016.

Y. Prabhu and M. Varma. FastXML: a fast, accurate and stable tree-classifier for extreme multi-label learning. In *KDD*, 2014.

P. Rai, C. Hu, R. Henao, and L. Carin. Large-scale bayesian multi-label learning via topic-based label embeddings. In *NIPS*, 2015.

A. P. Singh and G. J. Gordon. A Bayesian matrix factorization model for relational data. In *UAI*, 2010.

H.-F. Yu, P. Jain, P. Kar, and I. Dhillon. Large-scale multi-label learning with missing labels. In *ICML*, 2014.

H. Zhao, L. Du, and W. Buntine. Leveraging node attributes for incomplete relational data. In *ICML*, 2017a.

H. Zhao, L. Du, and W. Buntine. A word embeddings informed focused topic model. In *ACML*, 2017b.

H. Zhao, L. Du, W. Buntine, and G. Liu. MetaLDA: A topic model that efficiently incorporates meta information. In *ICDM*, 2017c.

M. Zhou. Infinite edge partition models for overlapping community detection and link prediction. In *AISTATS*, 2015.

M. Zhou, L. Hannah, D. Dunson, and L. Carin. Beta-negative binomial process and Poisson factor analysis. In *AISTATS*, 2012.

Chapter 8

Conclusion

In my PhD study, I have been mainly working on Bayesian latent factor modelling and inference problems for discrete data in the applications of text analysis, graph analysis, and multi-label learning, with a focus on leveraging meta-data and discovering structured latent values. The proposed approaches in my PhD research have achieved not only the state-of-the-art modelling accuracy but also excellent interpretability. This thesis elaborates on the details of my PhD research and the main content of the thesis can be summarised as follows:

- As my PhD research focuses on the theme of Bayesian latent factor models (BLFMs), Chapter 1 presents the general introduction of BLFMs, as well as the motivations and importance of using these models in practical applications. In addition, this chapter summarises the contributions from the perspective of different applications and lists the published papers in my PhD study.

- Chapter 2 covers the fundamental knowledge of Bayesian analysis, including the choices of data and prior distributions, conjugate priors, data augmentation techniques, and the basics of Bayesian inference with a focus on MCMC sampling. These fundamentals are essential for understanding the remaining chapters of the thesis. In addition, this chapter also elaborates on a unified framework of BLFMs, which serves as the basic framework of my PhD study.

- Based on Chapter 2, Chapter 3 comprehensively reviews the related works of BLFMs for discrete in the areas of text analysis with a focus on models with meta-data, short-topic models, deep/neural topic models; graph analysis with Bayesian graphical models; and multi-label learning with Bayesian graphical models.

- Chapter 4 describes my research work on extending the unified BLFM framework into graph analysis. The proposed model is able to effectively and efficiently leverage note attributes to improve the performance of link prediction and community detection in relational graph analysis, especially in the cases where a graph is highly incomplete.

- Chapter 5 presents the research work on topic modelling with meta-data, which can be viewed as an extension of the unified BLFM framework in the area of text analysis. Specifically, a general framework is proposed, which efficiently incorporates various types of meta-data such as document labels and word embeddings for discovering more interpretable topics from texts. The developed topic modelling framework can achieve better modelling performance as well as improved interpretability, especially for short texts such as tweets and news headlines.

- Chapter 6 presents the work on hierarchical topic discovery for text analysis, which are multi-layer extensions of the unified framework of BLFMs. The proposed models are able to significantly improve the intuitive understanding of fine-grained semantic structures of texts.
- Chapter 7 shows the details of my work on extending the basic framework of BLFMs into multi-label learning problems. The main novelty of the proposed model described in this chapter is the structure that leverages the sparsity of both the label and feature matrices, making the inference efficient for binary features.

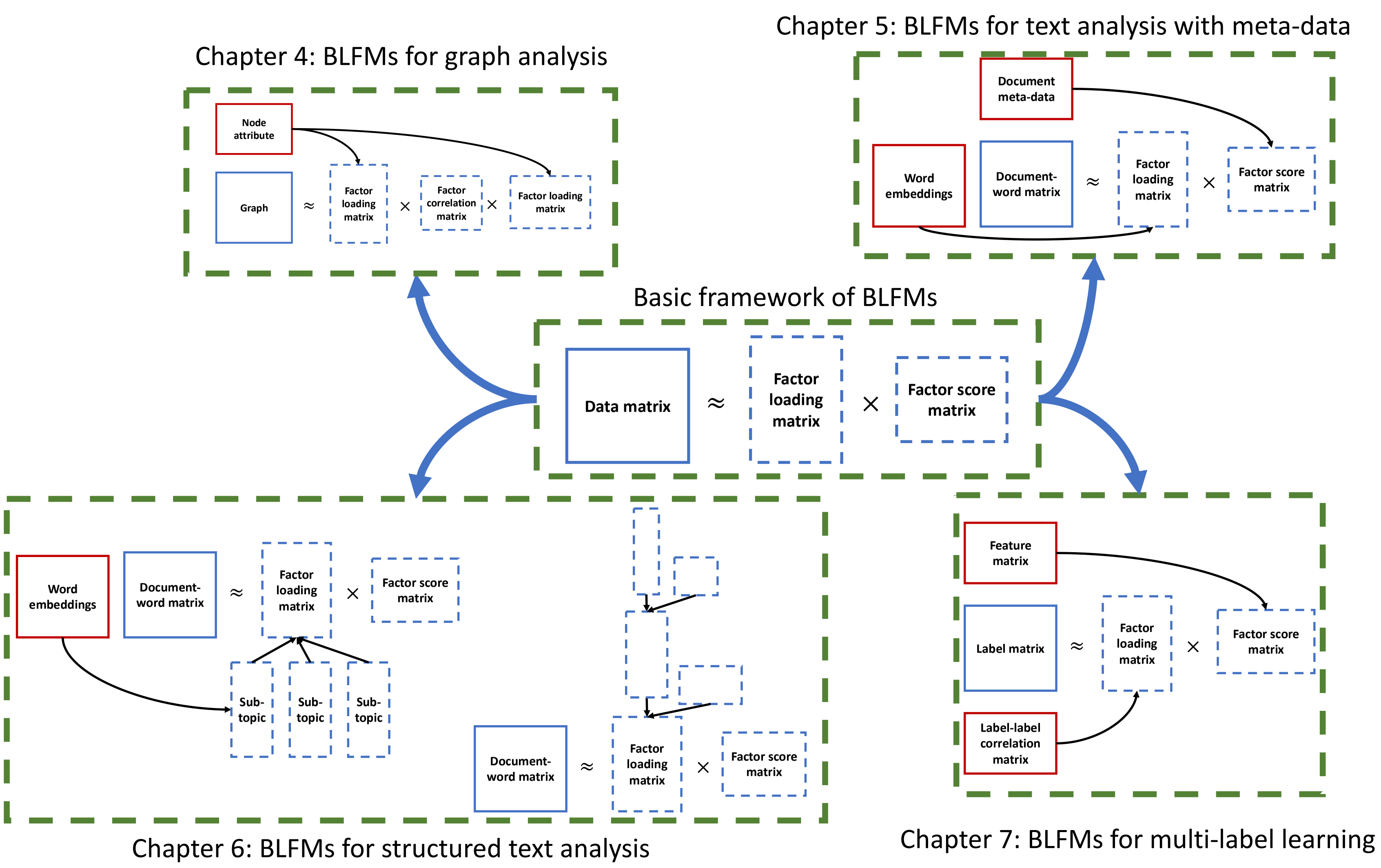


Figure 8.1: Summary of the comparisons between the proposed models and the unified framework of BLFMs.

Table 8.1: Comparisons of the proposed models in terms of target data, meta-data, sparsity, model structures.

Model	Target data	Meta-data	Structured model	Data sparsity	Meta-data sparsity
Zhao et al. [2017a] in Chapter 4	Adjacency matrix of an unweighted graph	Binary attributes of nodes	No	Yes	Yes
Zhao et al. [2017c, 2018a] in Chapter 5	Texts	Binary document and word features	No	Yes	Yes
Zhao et al. [2017b] in Chapter 5	Texts	Real-valued word embeddings	No	Yes	No
Zhao et al. [2018d] in Chapter 7	Label matrix in multi-label learning	Binary features	No	Yes	Yes
Zhao et al. [2018c] in Chapter 6	Texts	Real-valued word embeddings	Yes	Yes	No
Zhao et al. [2018b] in Chapter 6	Texts	N/A	Yes	Yes	N/A

In this chapter, I systemically show the connections and comparisons between the models presented in Chapter 4 to 7, details as follows:

- Figure 8.1 demonstrates how the models in the above chapters extend the unified framework of BLFMs introduced in Chapter 2. Specifically, the basic framework factorises a data matrix into two latent matrices and the proposed models extend it into various areas including graph analysis, text analysis, and multi-label learning, with the ability to incorporate meta-data and discovering hierarchical latent structures. Moreover, to tackle the associated inference problems, various data augmentation techniques have been adopted and data sparsity has been carefully taken into account.

- Table 8.1 shows the comparisons of the proposed models in the above chapters in terms of the types of target data, types of meta-data, whether a model uses structured latent variables, whether a model captures target data and meta-data sparsity.

- Based on Table 8.1, Table 8.2 further demonstrates the comparisons between the proposed models in terms of applications, assumptions, and constraints.

Now I re-summarise the major contributions of my PhD research. Recall that Chapter 1 introduces the contributions specific to individual applications including graph analysis, text analysis, and multi-label learning. Here the contributions are summarised in a different perspective, detailed as follows:

- **Meta-data incorporation:** In the areas of text and graph analysis, meta-data are usually accessible and able to serve as important supplementary information especially when the target data are sparse or incomplete. In this research, several BLFMs have been proposed for the effective and efficient incorporation of meta-data in text analysis, graph analysis, and multi-label learning. With the help of meta-data, the proposed approaches have achieved the state-of-the-art numerical performance on the tasks of text modelling, link prediction for relational graphs, and multi-label classification. In addition, the proposed methods are able to leverage the sparsity in target data and meta-data, obtaining improved efficiency.

- **Interpretability:** Interpretability is increasingly important in machine learning. The proposed BLFMs have made substantial contributions in Bayesian analysis for text and graph analysis. Specifically, the developed models with meta-data enjoy better interpretability by intuitively discovering the connections between meta-data and target data, which can be easily interpreted and visualised. Moreover, the latent structures in the proposed hierarchical models can be used to explain the structures

Table 8.2: Comparisons of the proposed models in terms of applications, assumptions, and constraints.

Model	Applications	Assumptions	Constraints
Zhao et al. [2017a] in Chapter 4	1. Unweighted relational graph analysis 2. Link prediction and community detection 3. Incorporation of meta-data of nodes in graphs	1. Nodes are sparsely connected 2. Meta-data of nodes is sparse 3. Assortativity: nodes of the same communities are more likely to be connected 4. Nodes with similar meta-data are more likely to be connected	1. Node meta-data needs to be binarised 2. Gibbs sampling may not be efficient for large data
Zhao et al. [2017c, 2018a] in Chapter 5	1. Topic analysis for documents 2. Feature extraction for downstream applications e.g.text classification and clustering 3. Short text topic analysis 4. Incorporation of document and word meta-data 5. Efficient and parallel implementation for large-scale data	1. Documents with similar meta-data are more likely to discuss similar topics 2. Words with similar meta-data are more likely to be assigned with similar topics	1. Document and word meta-data need to binarised
Zhao et al. [2017b] in Chapter 5	1. Topic analysis for documents 2. Feature extraction for downstream applications e.g.text classification and clustering 3. Short text topic analysis 4. Incorporation of real-valued word embeddings 5. Focusing analysis of topics and words	1. If the positions of two words are close in the embedding space, they are more likely to be assigned with similar topics 2. A topic is usually described by a subset of words in the vocabulary 3. A word usually describes a subset of topics	1. Inference of topic embeddings may not be efficient for large data
Zhao et al. [2018d] in Chapter 7	1. Multi-label classification 2. Incorporation of label-label cooccurences 3. Efficient implementation for binary feature	1. Label matrix is sparse 2. Binary feature is sparse	1. Features of samples need to be binarised 2. Gibbs sampling may not be efficient for large data
Zhao et al. [2018c] in Chapter 6	1. Topic analysis for documents 2. Feature extraction for downstream applications e.g.text classification and clustering 3. Short text topic analysis 4. Incorporation of real-valued word embeddings 5. Sub-topic structure discovery	1. An individual topic may not be semantically indivisible and it can be split into several sub-topics 2. Topics can mix the words which co-occur locally in the target corpus but are less semantically related in general 3. Word embeddings trained on a large external corpus can be used as the global semantics for sub-topic discovery	1. The external corpus for training word embeddings needs to be close to the target corpus, otherwise the word embeddings may provide less help 2. Gibbs sampling may not be efficient for large data
Zhao et al. [2018b] in Chapter 6	1. Topic analysis for documents 2. Feature extraction for downstream applications e.g.text classification and clustering 3. Short text topic analysis 4. Hierarchical topic structure discovery	1. In a topic hierarchy, topics in the higher layers are more general than those in the lower layers	1. Topics in the higher layers in a topic hierarchy with too many layers may be too general to recognise 2. Gibbs sampling may not be efficient for large data

of data, which provides a more intuitive way for understanding and visualising text and graph data.

- **Usability:** The proposed models have great potential in many applications. For example, the latent factors on texts by the approaches introduced in Chapter 5 and 6 can be used as the embeddings of documents, which can be fed into downstream applications such as document classification and clustering. Moreover, these text analysis approaches have shown substantial improvements in modelling short texts. To further assist the usability of this PhD research, I have released well-engineered code with constructive instructions on installation and reproduction of the results for the proposed models.

- **Scalability:** The proposed approaches intrinsically enjoy excellent efficiency because of the consideration of data sparsity, which is an important property in text and graph analysis, as well as in multi-label learning. In most of the proposed models, the computation only needs to be spent on the non-zero data, which saves a huge amount of training time. Furthermore, in the released code, efficient implementations have been used to assist scalability. For example, the inference algorithm in Zhao et al. [2017c] was carefully implemented with multi-thread programming, which is able to execute on supercomputers or clusters for large-scale data.

In addition to the constructive and detailed introduction to my PhD research, the contributions of the thesis includes:

- Providing a proper coverage of the background knowledge of Bayesian Analysis and the key techniques used in the proposed models.

- Providing a comprehensive review of the related works in the areas of topic modelling, graph analysis, and multi-label learning, including the critical comparisons of existing methods, open problems, and popular research directions.

In the area of text analysis with topic models, further studies involve the following directions:

- Incorporating other types of word meta-data, such as part-of-speech tags of words and information from knowledge graphs like WordNet [Fellbaum, 2012]. Note that the proposed models with meta-data are able to incorporate meta-data formulated into vectors/matrices. This setting may not be suitable for other types of word meta-data than embeddings, such as the above examples. Therefore, a possible future direction for topic models with meta-data is to develop proper model structures for more complex meta-data.

- Discovering topic structures jointly in multi-domain corpora or multilingual corpora. Multi-domain topic modelling is about discovering common and domain-specific topics for comparing document in multi-domain corpora [Chen and Liu, 2014], while multilingual topic models discover topics from multilingual corpora [Boyd-Graber and Blei, 2009]. It would be interesting to apply the topic structure models presented in Chapter 6 to the above two kinds of datasets to intuitively understand the distinctions and connections between multiple corpora.

- Developing scalable inference algorithms for the proposed models on larger datasets. The proposed models enjoy better modelling accuracy and interpretability, but they also add extra model complexity, which requires more efficient inference schemes. Although data and meta-data sparsity is leveraged in the proposed models, it is necessary to study how this property can be used in algorithms including variational inference and SGMCMC.

In the area of graph analysis, future research directions can be on leveraging other types of meta-data such as link attributes and applying the developed model on multi-relational graphs such as knowledge graphs [Hu et al., 2016b]. In the area of multi-label learning, further studies are needed on the ways of leveraging the data sparsity for non-binary features, and better dealing with the missing label problem, which are open problems in this area.

Bibliography

Ayan Acharya, Dean Teffer, Jette Henderson, Marcus Tyler, Mingyuan Zhou, and Joydeep Ghosh. Gamma process Poisson factorization for joint modeling of network and documents. In *ECML and PKDD*, pages 283–299, 2015.

Amr Ahmed, Liangjie Hong, and Alexander Smola. Nested Chinese restaurant franchise process: Applications to user tracking and document modeling. In *ICML*, pages 1426–1434, 2013.

Edo M Airoldi, David M Blei, Stephen E Fienberg, and Eric P Xing. Mixed membership stochastic blockmodels. In *NIPS*, pages 33–40, 2009.

Nikolaos Aletras and Mark Stevenson. Evaluating topic coherence using distributional semantics. In *International Conference on Computational Semantics*, pages 13–22, 2013.

Anima Anandkumar, Dean P Foster, Daniel J Hsu, Sham M Kakade, and Yi-Kai Liu. A spectral algorithm for latent Dirichlet allocation. In *NIPS*, pages 917–925, 2012.

David Andrzejewski, Xiaojin Zhu, Mark Craven, and Benjamin Recht. A framework for incorporating general domain knowledge into latent Dirichlet allocation using first-order logic. In *IJCAI*, pages 1171–1177, 2011.

Cedric Archambeau, Balaji Lakshminarayanan, and Guillaume Bouchard. Latent IBP compound Dirichlet allocation. *IEEE TPAMI*, 37(2):321–333, 2015.

Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models–going beyond SVD. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 1–10, 2012.

R. Babbar and B. Schölkopf. DiSMEC-distributed sparse machines for extreme multi-label classification. In *WSDM*, 2017.

Wei Bi and James T Kwok. Multilabel classification with label correlations and missing labels. In *AAAI*, 2014.

David Blackwell, James B MacQueen, et al. Ferguson distributions via Pólya urn schemes. *The annals of statistics*, 1(2):353–355, 1973.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet allocation. *JMLR*, 3:993–1022, 2003.

David M Blei, Thomas L Griffiths, and Michael I Jordan. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57(2):7, 2010.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795, 2013.

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. Generating sentences from a continuous space. In *CoNLL*, pages 10–21, 2016.

Jordan Boyd-Graber and David M Blei. Multilingual topic models for unaligned text. In *UAI*, pages 75–82, 2009.

John Canny. Gap: A factor model for discrete data. In *SIGIR*, pages 122–129, 2004.

Dallas Card, Chenhao Tan, and Noah A Smith. Neural models for documents with metadata. In *ACL*, pages 2031–2040, 2018.

François Caron and Emily B Fox. Sparse graphs using exchangeable random measures. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(5): 1295–1366, 2017.

Jonathan Chang and David Blei. Relational topic models for document networks. In *AISTATS*, pages 81–88, 2009.

Changyou Chen, Lan Du, and Wray Buntine. Sampling table configurations for the hierarchical Poisson-Dirichlet process. In *ECML*, pages 296–311, 2011.

Jianfei Chen, Kaiwei Li, Jun Zhu, and Wenguang Chen. WarpLDA: A cache efficient O(1) algorithm for latent dirichlet allocation. *VLDB*, 9(10):744–755, 2016.

Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient Hamiltonian Monte Carlo. In *ICML*, pages 1683–1691, 2014.

Zhiyuan Chen and Bing Liu. Topic modeling using topics from many domains, lifelong learning and big data. In *ICML*, pages 703–711, 2014.

Aaron Clauset, Cristopher Moore, and Mark EJ Newman. Hierarchical structure and the prediction of missing links in networks. *Nature*, 453(7191):98–101, 2008.

Yulai Cong, Bo Chen, Hongwei Liu, and Mingyuan Zhou. Deep latent Dirichlet allocation with topic-layer-adaptive stochastic gradient Riemannian MCMC. In *ICML*, pages 864–873, 2017.

Rajarshi Das, Manzil Zaheer, and Chris Dyer. Gaussian LDA for topic models with word embeddings. In *ACL*, pages 795–804, 2015.

Jesse Davis and Mark Goadrich. The relationship between Precision-Recall and ROC curves. In *ICML*, pages 233–240, 2006.

Xuhui Fan, Richard Yi Da Xu, Longbing Cao, and Yin Song. Learning nonparametric relational models by conjugately incorporating node information in a network. *IEEE transactions on cybernetics*, 47(3):589–599, 2017.

Christiane Fellbaum. Wordnet. *The Encyclopedia of Applied Linguistics*, 2012.

Thomas S Ferguson. A Bayesian analysis of some nonparametric problems. *The annals of statistics*, pages 209–230, 1973.

Xianghua Fu, Ting Wang, Jing Li, Chong Yu, and Wangwang Liu. Improving distributed word representation and topic model by word-topic mixture model. In *ACML*, pages 190–205, 2016.

Zhe Gan, Changyou Chen, Ricardo Henao, David Carlson, and Lawrence Carin. Scalable deep Poisson factor analysis for topic modeling. In *ICML*, pages 1823–1832, 2015a.

Zhe Gan, R. Henao, D. Carlson, and Lawrence Carin. Learning deep sigmoid belief networks with data augmentation. In *AISTATS*, pages 268–276, 2015b.

Abhilash Gaure, Aishwarya Gupta, Vinay Kumar Verma, and Piyush Rai. A probabilistic framework for zero-shot multi-label learning. In *UAI*, volume 1, page 3.

Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE TPAMI*, (6):721–741, 1984.

Zoubin Ghahramani and T.L. Griffiths. Infinite latent feature models and the Indian buffet process. In *NIPS*, pages 475–482, 2006.

Eva Gibaja and Sebastián Ventura. A tutorial on multilabel learning. *ACM Computing Survey*, 2015.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

Prem K Gopalan, Sean Gerrish, Michael Freedman, David M Blei, and David M Mimno. Scalable inference of overlapping communities. In *NIPS*, pages 2249–2257, 2012.

Prem K Gopalan, Chong Wang, and David Blei. Modeling overlapping communities with node popularities. In *NIPS*, pages 2850–2858, 2013.

Prem K Gopalan, Laurent Charlin, and David Blei. Content-based recommendations with Poisson factorization. In *NIPS*, pages 3176–3184, 2014.

Thomas L. Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5228–5235, 2004.

WK HASTINGS. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

Ricardo Henao, Zhe Gan, James Lu, and Lawrence Carin. Deep Poisson factor modeling. In *NIPS*, pages 2800–2808, 2015.

Tue Herlau, Mikkel N Schmidt, Lars Kai Hansen, et al. Detecting hierarchical structure in networks. In *International Workshop on Cognitive Information Processing*, pages 1–6, 2012.

Qirong Ho, Ankur Parikh, Le Song, and Eric Xing. Multiscale community blockmodel for network exploration. In *AISTATS*, pages 333–341, 2011.

Matthew D Hoffman, David M Blei, Chong Wang, and John Paisley. Stochastic variational inference. *JMLR*, 14(1):1303–1347, 2013.

Liangjie Hong and Brian D Davison. Empirical study of topic modeling in Twitter. In *The First Workshop on Social Media Analytics*, pages 80–88, 2010.

Changwei Hu, Piyush Rai, and Lawrence Carin. Non-negative matrix factorization for discrete data with hierarchical side-information. In *AISTATS*, pages 1124–1132, 2016a.

Changwei Hu, Piyush Rai, and Lawrence Carin. Topic-based embeddings for learning from large knowledge graphs. In *AISTATS*, pages 1133–1141, 2016b.

Himanshu Jain, Yashoteja Prabhu, and Manik Varma. Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In *SIGKDD*, pages 935–944, 2016.

Vikas Jain, Nirbhay Modhe, and Piyush Rai. Scalable generative models for multi-label learning with missing labels. In *ICML*, pages 1636–1644, 2017.

Ou Jin, Nathan N Liu, Kai Zhao, Yong Yu, and Qiang Yang. Transferring topical knowledge from auxiliary long texts for short text clustering. In *CIKM*, pages 775–784, 2011.

Ashish Kapoor, Raajay Viswanathan, and Prateek Jain. Multilabel classification using Bayesian compressed sensing. In *NIPS*, pages 2645–2653, 2012.

Charles Kemp, Joshua B Tenenbaum, Thomas L Griffiths, Takeshi Yamada, and Naonori Ueda. Learning systems of concepts with an infinite relational model. In *AAAI*, volume 3, page 5, 2006.

Dae Il Kim, Michael C Hughes, and Erik B Sudderth. The nonparametric metadata dependent relational model. In *ICML*, pages 1411–1418, 2012a.

Dae Il Kim, Prem K Gopalan, David Blei, and Erik Sudderth. Efficient online inference for Bayesian nonparametric relational models. In *NIPS*, pages 962–970, 2013.

Dongwoo Kim and Alice Oh. Hierarchical Dirichlet scaling process. *Machine Learning*, 106(3):387–418, 2017.

Joon Hee Kim, Dongwoo Kim, Suin Kim, and Alice Oh. Modeling topic hierarchies with the recursive Chinese restaurant process. In *CIKM*, pages 783–792, 2012b.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Phaedon-Stelios Koutsourelakis and Tina Eliassi-Rad. Finding mixed-memberships in social networks. In *AAAI Spring Symposium: Social Information Processing*, pages 48–53, 2008.

Rahul Krishnan, Dawen Liang, and Matthew Hoffman. On the challenges of learning with inference networks on sparse, high-dimensional data. In *AISTATS*, pages 143–151, 2018.

John D Lafferty and David M Blei. Correlated topic models. In *NIPS*, pages 147–154, 2006.

Jey Han Lau, David Newman, and Timothy Baldwin. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *EACL*, pages 530–539, 2014.

Aaron Q Li, Amr Ahmed, Sujith Ravi, and Alexander J Smola. Reducing the sampling complexity of topic models. In *SIGKDD*, pages 891–900, 2014.

Chenliang Li, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. Topic modeling for short texts with auxiliary word embeddings. In *SIGIR*, pages 165–174, 2016.

Chenliang Li, Yu Duan, Haoran Wang, Zhiqian Zhang, Aixin Sun, and Zongyang Ma. Enhancing topic modeling for short texts with auxiliary word embeddings. *ACM Transactions on Information Systems*, 36(2):11, 2017.

Wei Li and Andrew McCallum. Pachinko allocation: DAG-structured mixture models of topic correlations. In *ICML*, pages 577–584, 2006.

Dawen Liang, Rahul G Krishncan, Matthew D Hoffman, and Tony Jebara. Variational autoencoders for collaborative filtering. In *WWW*, pages 689–698, 2018.

A. Lijoi and I. Prünster. Models beyond the Dirichlet process. In N.L. Hjort, C. Holmes, P. Müller, and S.G. Walker, editors, *Bayesian Nonparametrics*, pages 80–135. Cambridge University Press, 2010.

Kar Wai Lim and Wray Buntine. Bibliographic analysis on research publications using authors, categorical labels and the citation network. *Machine Learning*, 103(2):185–213, 2016.

Kar Wai Lim, Changyou Chen, and Wray L. Buntine. Twitter-Network Topic Model: A full Bayesian treatment for social network and text modeling. In *NIPS: Topic Models Workshop*, pages 1–5, 2013.

Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. Topical word embeddings. In *AAAI*, pages 2418–2424, 2015.

Yi-An Ma, Nicholas J Foti, and Emily B Fox. Stochastic gradient MCMC methods for hidden Markov models. In *ICML*, pages 2265–2274, 2017.

Jon D Mcauliffe and David M Blei. Supervised topic models. In *NIPS*, pages 121–128, 2008.

Rishabh Mehrotra, Scott Sanner, Wray Buntine, and Lexing Xie. Improving LDA topic models for microblogs via tweet pooling and automatic labeling. In *SIGIR*, pages 889–892, 2013.

Thomas Mensink, Efstratios Gavves, and Cees GM Snoek. Costa: Co-occurrence statistics for zero-shot classification. In *CVPR*, pages 2441–2448, 2014.

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

Yishu Miao, Lei Yu, and Phil Blunsom. Neural variational inference for text processing. In *ICML*, pages 1727–1736, 2016.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. Discovering discrete latent topics with neural variational inference. In *ICML*, pages 2410–2419, 2017.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionally. In *NIPS*, pages 3111–3119, 2013.

Kurt Miller, Michael I Jordan, and Thomas L Griffiths. Nonparametric latent feature models for link prediction. In *NIPS*, pages 1276–1284, 2009.

David Mimno and Andrew McCallum. Topic models conditioned on arbitrary features with Dirichlet-multinomial regression. In *UAI*, pages 411–418, 2008.

Paul Mineiro and Nikos Karampatziakis. Fast label embeddings via randomized linear algebra. In *ECML*, 2015.

Andriy Mnih and Ruslan R Salakhutdinov. Probabilistic matrix factorization. In *NIPS*, pages 1257–126f4, 2008.

Morten Mørup and Mikkel N Schmidt. Bayesian community detection. *Neural Computation*, 24(9):2434–2456, 2012.

Morten Mørup, Mikkel N Schmidt, and Lars Kai Hansen. Infinite multiple membership relational modeling for complex networks. In *IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–6, 2011.

Radford M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2:113–162, 2011.

Dat Quoc Nguyen, Richard Billingsley, Lan Du, and Mark Johnson. Improving topic models with latent feature word representations. *Transactions of the Association for Computational Linguistics*, 3:299–313, 2015.

Kamal Nigam, Andrew Kachites McCallum, Sebastian Thrun, and Tom Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39 (2-3):103–134, 2000.

K. Nowicki and T.A.B. Snijders. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001.

John Paisley, Chong Wang, David M Blei, and Michael I Jordan. Nested hierarchical Dirichlet processes. *IEEE TPAMI*, 37(2):256–270, 2015.

Konstantina Palla, David Knowles, and Zoubin Ghahramani. An infinite latent attribute model for network data. *ICML*, pages 395–402, 2012.

Konstantina Palla, David A Knowles, and Zoubin Ghahramani. Relational learning and network modelling using infinite latent attribute models. *IEEE TPAMI*, 37(2):462–474, 2015.

Fragkiskos Papadopoulos, Maksim Kitsak, M Ángeles Serrano, Marián Boguná, and Dmitri Krioukov. Popularity versus similarity in growing networks. *Nature*, 489(7417):537, 2012.

Sam Patterson and Yee Whye Teh. Stochastic gradient Riemannian Langevin dynamics on the probability simplex. In *NIPS*, pages 3102–3110, 2013.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *EMNLP*, pages 1532–1543, 2014.

Jim Pitman et al. Combinatorial stochastic processes. Technical report, Technical Report 621, Dept. Statistics, UC Berkeley, 2002. Lecture notes for St. Flour course, 2002.

Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349, 2013.

Yashoteja Prabhu and Manik Varma. FastXML: A fast, accurate and stable tree-classifier for extreme multi-label learning. In *SIGKDD*, 2014.

Xiaojun Quan, Chunyu Kit, Yong Ge, and Sinno Jialin Pan. Short and sparse text topic modeling via self-aggregation. In *IJCAI*, pages 2270–2276, 2015.

Piyush Rai, Changwei Hu, Ricardo Henao, and Lawrence Carin. Large-scale Bayesian multi-label learning via topic-based label embeddings. In *NIPS*, pages 3222–3230, 2015.

Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In *EMNLP*, pages 248–256, 2009.

Daniel Ramage, Christopher D Manning, and Susan Dumais. Partially labeled topic models for interpretable text mining. In *SIGSIGKDD*, pages 457–465, 2011.

Rajesh Ranganath, Linpeng Tang, Laurent Charlin, and David Blei. Deep exponential families. In *AISTATS*, pages 762–771, 2015.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, pages 1278–1286, 2014.

Michael Röder, Andreas Both, and Alexander Hinneburg. Exploring the space of topic coherence measures. In *WSDM*, pages 399–408, 2015.

Daniel M Roy and Yee W Teh. The Mondrian process. In *NIPS*, pages 1377–1384, 2009.

Jayaram Sethuraman. A constructive definition of Dirichlet priors. *Statistica sinica*, pages 639–650, 1994.

Tian Shi, Kyeongpil Kang, Jaegul Choo, and Chandan K Reddy. Short-text topic modeling via non-negative matrix factorization enriched with local word-context correlations. In *WWW*, pages 1105–1114, 2018.

Akash Srivastava and Charles Sutton. Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*, 2017.

Yee W Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Sharing clusters among related groups: Hierarchical Dirichlet processes. In *NIPS*, pages 1385–1392, 2005.

Y.W. Teh, M.I. Jordan, M.J. Beal, and D.M. Blei. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2012.

H.M. Wallach, I. Murray, R. Salakhutdinov, and D. Mimno. Evaluation methods for topic models. In L. Bottou and M. Littman, editors, *ICML*, 2009.

Y.J. Wang and G.Y. Wong. Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, 82(397):8–19, 1987.

Sinead Williamson, Chong Wang, Katherine A Heller, and David M Blei. The IBP compound Dirichlet process and its application to focused topic modeling. In *ICML*, pages 1151–1158, 2010.

Robert L Wolpert, Merlise A Clyde, Chong Tu, et al. Stochastic expansions using continuous dictionaries: Lévy adaptive regression kernels. *The Annals of Statistics*, 39(4): 1916–1962, 2011.

Junyu Xuan, Jie Lu, Guangquan Zhang, Richard Yi Da Xu, and Xiangfeng Luo. A Bayesian nonparametric model for multi-label learning. *Machine Learning*, 106(11): 1787–1815, 2017.

Guangxu Xun, Vishrawas Gopalakrishnan, Fenglong Ma, Yaliang Li, Jing Gao, and Aidong Zhang. Topic discovery for short texts using word embeddings. In *ICDM*, pages 1299–1304, 2016.

Xiaohui Yan, Jiafeng Guo, Yanyan Lan, and Xueqi Cheng. A biterm topic model for short texts. In *WWW*, pages 1445–1456, 2013.

Bo Yang and Xuehua Zhao. On the scalable learning of stochastic blockmodel. In *AAAI*, pages 360–366, 2015.

Yang Yang, Feifei Wang, Junni Zhang, Jin Xu, and S Yu Philip. A topic model for co-occurring normal documents and short texts. *WWW*, 21(2):487–513, 2018.

Yi Yang, Doug Downey, and Jordan Boyd-Graber. Efficient methods for incorporating knowledge into topic models. In *EMNLP*, pages 308–317, 2015.

Limin Yao, David Mimno, and Andrew McCallum. Efficient methods for topic model inference on streaming document collections. In *SIGSIGKDD*, pages 937–946, 2009.

Jianhua Yin and Jianyong Wang. A Dirichlet multinomial mixture model-based approach for short text clustering. In *SIGSIGKDD*, pages 233–242. ACM, 2014.

Hsiang-Fu Yu, Prateek Jain, Purushottam Kar, and Inderjit Dhillon. Large-scale multi-label learning with missing labels. In *ICML*, pages 593–601, 2014.

Hsiang-Fu Yu, Cho-Jui Hsieh, Hyokun Yun, SVN Vishwanathan, and Inderjit S Dhillon. A scalable asynchronous distributed algorithm for topic modeling. In *WWW*, pages 1340–1350, 2015.

Jinhui Yuan, Fei Gao, Qirong Ho, Wei Dai, Jinliang Wei, Xun Zheng, Eric Po Xing, Tie-Yan Liu, and Wei-Ying Ma. LightLDA: Big topic models on modest computer clusters. In *WWW*, pages 1351–1361, 2015.

Jia Zeng, William K Cheung, and Jiming Liu. Learning topic models by belief propagation. *IEEE TPAMI*, 35(5):1121–1134, 2013.

Hao Zhang, Bo Chen, Dandan Guo, and Mingyuan Zhou. WHAI: Weibull hybrid autoencoding inference for deep topic modeling. 2018.

He Zhao, Lan Du, and Wray Buntine. Leveraging node attributes for incomplete relational data. In *ICML*, pages 4072–4081, 2017a.

He Zhao, Lan Du, and Wray Buntine. A word embeddings informed focused topic model. In *ACML*, pages 423–438, 2017b.

He Zhao, Lan Du, Wray Buntine, and Gang Liu. MetaLDA: A topic model that efficiently incorporates meta information. In *ICDM*, pages 635–644, 2017c.

He Zhao, Lan Du, Wray Buntine, and Gang Liu. Leveraging external information in topic modelling. *Knowledge and Information Systems*, pages 1–33, 2018a.

He Zhao, Lan Du, Wray Buntine, and Mingyuan Zhou. Dirichlet belief networks for topic structure learning. In *NeurIPS*, pages 7966–7977, 2018b.

He Zhao, Lan Du, Wray Buntine, and Mingyuan Zhou. Inter and intra topic structure learning with word embeddings. In *ICML*, pages 5887–5896, 2018c.

He Zhao, Piyush Rai, Lan Du, and Wray Buntine. Bayesian multi-label learning with sparse features and labels, and label co-occurrences. In *AISTATS*, pages 1943–1951, 2018d.

He Zhao, Lan Du, Guanfeng Liu, and Wray Buntine. Leveraging meta information in short text aggregation. In *ACL*, 2019a.

He Zhao, Piyush Rai, Lan Du, Wray Buntine, and Mingyuan Zhou. Variational autoencoders for sparse and overdispersed discrete data. *arXiv preprint arXiv:1905.00616*, 2019b.

MingYuan Zhou. Infinite edge partition models for overlapping community detection and link prediction. In *AISTATS*, pages 1135–1143, 2015.

Mingyuan Zhou. Parsimonious Bayesian deep networks. In *NIPS*, pages 3190–3200, 2018.

Mingyuan Zhou and Lawrence Carin. Negative binomial process count and mixture modeling. *IEEE TPAMI*, 37(2):307–320, 2015.

Mingyuan Zhou, Lauren Hannah, David B Dunson, and Lawrence Carin. Beta-negative binomial process and Poisson factor analysis. In *AISTATS*, pages 1462–1471, 2012a.

Mingyuan Zhou, Lingbo Li, David Dunson, and Lawrence Carin. Lognormal and gamma mixed negative binomial regression. In *ICML*, volume 2012, page 1343, 2012b.

Mingyuan Zhou, Yulai Cong, and Bo Chen. The Poisson gamma belief network. In *NIPS*, pages 3043–3051, 2015.

Mingyuan Zhou, Yulai Cong, and Bo Chen. Augmentable gamma belief networks. *JMLR*, 17(163):1–44, 2016.

Yaojia Zhu, Xiaoran Yan, Lise Getoor, and Cristopher Moore. Scalable text and link analysis with mixed-topic link models. In *SIGKDD*, pages 473–481, 2013.

Yuan Zuo, Junjie Wu, Hui Zhang, Hao Lin, Fei Wang, Ke Xu, and Hui Xiong. Topic modeling of short texts: A pseudo-document view. In *SIGSIGKDD*, pages 2105–2114, 2016a.

Yuan Zuo, Jichang Zhao, and Ke Xu. Word network topic model: A simple but general solution for short and imbalanced texts. *Knowledge and Information Systems*, 48(2): 379–398, 2016b.