A study on dual topic models and bayesian topic model inference

Rugeles, Daniel

2020

Rugeles, D. (2020). A study on dual topic models and bayesian topic model inference.
Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/145117>

<https://doi.org/10.32657/10356/145117>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0
International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 12:55:14 SGT

A Study on Dual Topic Models and Bayesian Topic Model Inference.

Daniel Rugeles

Supervisor: Prof. Gao Cong

Dr. Zhen Hai

School of Computer Science and Engineering

Nanyang Technological University

A thesis submitted to the Nanyang Technological University in partial
fulfillment of the requirement for the degree of PhD in Computer
Engineering

January 2021

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

May 3, 2019

A handwritten signature in dark ink that reads "DANIEL RUGELES". The signature is written in a cursive, slightly slanted style. The letters are connected, and the overall appearance is that of a personal signature.

Date

Daniel Rugeles

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

May 3, 2019

Date

Prof.Cong Gao

Authorship Attribution Statement

This thesis contains material from one paper accepted at one conference in which I am listed as an author.

Chapter 4 is published as Rugeles, D., Zhao, K., Cong, G., Dash, M., and Krishnaswamy, S. (2017). Bi- clustering: An application of dual topic models. In Proceedings of the 2017 SIAM International Conference on Data Mining, Houston, Texas, USA, April 27-29, 2017., pages 453–461

The contributions of the co-authors are as follows:

- Prof. Cong Gao and I reviewed the literature and formulated the key research problems.
- I designed and implemented our proposed model at the School of Computer Science and Engineering and the Institute for Infocomm Research under weekly supervision of Prof. Cong Gao and Dr. Manoranjan.
- Dr. Zhao assisted in the revision of the inference and generative process of the model.
- Dr. Manoranjan conducted the revision of the equations used to derive our proposed dual topics to bicluster model (DT2B), as well as the equations used to evaluate the models using predictive perplexity.

- Prof. Cong Gao and I reviewed the explanation of why DT2B works and why it generalizes the latent Dirichlet allocation and the author-topic model.
- Dr. Krishnaswamy assisted in the collection of singtel data and provided guidance in the experimental settings as well as the interpretation of the results.
- I prepared the manuscript drafts. The manuscripts were revised and edited with the help of all the authors.

May 3, 2019

A handwritten signature in dark ink that reads "DANIEL RUGELES". The signature is written in a cursive, slightly slanted style.

Date

Daniel Rugeles

Acknowledgements

I express my gratefulness for being part of the teams under the direction of Prof. Gao Cong at Nanyang Technological University, and Dr. Hai Zhen, Research Scientist at A*STAR. They have helped me improve my research skills by instructing me on how to apply critical thinking to produce state of the art contributions and how to scientifically conduct experiments. It is my pleasure to study and research under their supervision.

This work would not have been possible without the financial support granted by the Singapore International Graduate Award (SINGA). I am especially indebted to Dr. Manoranjan Dash, Senior Research Fellow at the National University of Singapore, and Dr. Shonali Krishnaswamy, CTO at AIDA Technologies Pte Ltd, and Professor at Swinburne University of Technology who provided guidance, academic feedback and revisions on my manuscripts, and who worked actively to help develop my ideas.

I am grateful to all of those with whom I have had the pleasure to work during this and other related projects. With their help, it has been possible to evolve in my research project. I specially thank Dr. Quan Yuan, Dr. Kaiqi Zhao, and Tay Yi from DMAL at NTU, and Dr. Joao Gomes, Dr. Nguyen Minh Nhut from A*STAR.

Nobody has been more important to me in the pursuit of this project than the members of my family and those close friends who I consider as family. Their life lessons are invaluable tools that combined with their love provided me unending inspiration and perseverance. I wish to thank my father Cayetano Rugeles, my mother Eneida Barbosa, the Schwartz family, the Eberlein family and my personal friends Sergio, Alvaro, Juan Felipe, Kshitij, and Eric.

Abstract

Topic models have been shown to be a powerful tool for organizing large collections of unstructured data, including text, images and location data. The essence of the topic model is to discover hidden clusters that summarize the relationship between two discrete random variables, such as documents and words in text data, or users and places in location data. In the former case, a topic typically refers to a cluster of semantically related words and the documents are described using the extracted topics.

Existing work on unsupervised clustering analysis primarily focuses on grouping one single type of random variable. However, applications in microarray and location data have proven that mutually clustering two kinds of random variables may be beneficial. This process is known as biclustering. Previous approaches to biclustering typically rely on a strategy of hard assignment of the data to the extracted clusters, known as hard clustering. In contrast, in this thesis we study how we can leverage topic models to present a novel strategy of soft clustering for biclustering problems. One key benefit of soft assignments over hard assignments is that the former is able to identify overlapping biclusters which is an inherent characteristic of many real-world applications [4]. Moreover, learning of topic models has been recognized as one of the most fundamental tasks. We also focus on the development of novel inference algorithms that can extract topics with an improved computational efficiency and predictive power. As we know, the sequential nature of the inference of topic models does not allow a straightforward parallel implementation. Existing distributed implementations approximate the results

of single threaded algorithms and may lower the quality of the extracted topics. Thus, we also study how to implement our inference algorithm using multiple cores without affecting the quality of the hidden topic detection.

In particular, the first part of the thesis treats LDA as a soft-clustering algorithm and then it introduces the dual LDA model that can be used to obtain complementary clusters on the same dataset. For example, location data contains the information of all the places visited by a set of users. On this dataset, LDA assigns the users to the clusters and the clusters represent probability distributions over the set locations. In contrast, the dual LDA assigns the locations to the clusters and the clusters represent probability distributions over the set of users. LDA and the dual LDA cluster one dataset from two different perspectives. We then propose to combine the two LDA models into a single model that summarizes the given dataset using soft-assignment for biclusters. We demonstrate applicability of the novel model in text data as well as microarray data. In our study, we show that our model is able to reveal a greater number of high quality biclusters from micro array data than established bicluster algorithms. This result also holds for synthetic datasets. In addition, we show that the proposed model can improve other topic models in a reviewer recommendation application.

In the second part of the thesis, we shift our attention to the learning of topic models. A popular inference approach to topic models is collapsed Gibbs sampling (CGS), which typically samples a single latent topic label for each observed document-word pair. First, we extend CGS and propose to assign a compound distribution over the possible assignments for a given document-word pair. The new approach, called recursively compound allocation (RCA), is able to leverage the benefit of soft-topic model assignments and achieve improved generalization performance. Then, we study the relationship between RCA and the state augmentation methods and we derive a

new generic deterministic inference method for learning a family of probabilistic topic models. One key benefit of the proposed method lies in its deterministic nature, which largely improves the running efficiency as well as predictive perplexity for inference.

We use well-known metrics to show how our proposed inference methods are able to extract topics with higher quality. In addition, our inference algorithm also shows higher predictive power in real world datasets as evidenced by evaluation based on the predictive perplexity using four topic models, two text datasets and a movie dataset. One of the evaluated topic models corresponds to our proposed topic model for biclustering. We also evaluate the impact on the predictive perplexity as we increase the number of cores as well as the speed of the algorithm. In both scenarios, we can clearly observe improvement of the proposed inference algorithm over the state of the art algorithm. Finally, we demonstrate that the increase in predictive power can be translated to an improvement in a real world application, i.e., we choose the popular document classification task.

Table of contents

List of figures	xiv
List of tables	xvii
1 Introduction	1
1.1 Background	1
1.2 Research Problems	2
1.3 Proposed Solutions	4
1.4 Contributions of this Thesis	6
1.5 Summary	8
2 Literature Review	9
2.1 Topic Models and its Applications	12
2.1.1 N-gram topic models	14
2.1.2 Bayesian non-parametric topic models	15
2.1.3 Correlated topic models	16
2.1.4 Temporal dynamics of the topic models	16
2.1.5 Extending LDA with additional variables	18
2.2 Biclustering	19
2.2.1 Biclustering	19
2.2.2 Biclustering structures	20

2.2.3	Biclustering types	20
2.2.4	Existing biclustering algorithms	21
2.3	Inference of Topic Models	23
2.3.1	Model-based inference methods	23
2.3.2	Family-based inference methods	25
2.4	Summary	27
3	Preliminaries	28
3.1	Probabilistic Bayesian Models	29
3.1.1	Topic models	31
3.1.2	Categorical mixture model	31
3.1.3	Latent Dirichlet allocation	32
3.1.4	Biclustering	34
3.1.5	Extensions on LDA	38
3.2	Gibbs Sampling Inference	39
3.2.1	Convergence of Gibbs sampling	40
3.2.2	Collapsed Gibbs sampling	42
3.2.3	Parameter estimation of the latent Dirichlet allocation	45
3.3	Summary	46
4	Dual Topic Models for Biclustering	49
4.1	Problem Definition	49
4.2	LDA and the Dual LDA	53
4.3	Dual Topics to Bicluster (DT2B)	54
4.3.1	Intuitions	54
4.3.2	Using CMM to implement intuitions 1 and 2	55

4.3.3	Defining the bivariate categorical distribution to implement intuition 3	56
4.3.4	Model representation	58
4.4	Generative and Inference Process	59
4.5	Connection with LDA and its Dual	61
4.5.1	Inference of LDA	62
4.5.2	Comparison with the inference equation of DT2B	62
4.6	Bicluster Extraction using DT2B	63
4.7	Why does DT2B work?	65
4.7.1	Topic influence between a pair of records	65
4.7.2	Mutual influence for the records in the same biclique	66
4.7.3	Applications and experimental evaluation	67
4.7.4	Datasets	69
4.7.5	Biclustering on synthetic data	70
4.7.6	Biclustering in microarray data	72
4.7.7	Reviewer recommendation.	75
4.7.8	User and place classification	77
4.7.9	Dimensionality reduction	80
4.7.10	Dual topics visualization	82
4.8	Summary	86
5	Inference of Topic Models	87
5.1	Recursively Compound Allocation	89
5.1.1	Principle of recursive compounding	89
5.1.2	Proof for convergence of RCA	97
5.1.3	Relation with mean field variational inference	99
5.2	Heron Inference	101

5.2.1	State augmentation methods	101
5.2.2	Maximization of the latent state replication	102
5.2.3	Convergence properties of Gibbs sampling	106
5.2.4	Resolving the sampling of repeated tuples	109
5.3	Contribution to the Inference of Topic Models	111
5.4	Experimental Evaluation	113
5.4.1	Datasets	113
5.4.2	Predictive perplexity of topic models	115
5.4.3	Hyperparameter sensitivity	119
5.4.4	Learning time	120
5.4.5	Automatic topic coherence	122
5.4.6	Applying Heron inference to DT2B	124
5.4.7	Document classification	126
5.5	Summary	131
6	Conclusion and Future Work	132
6.1	Conclusion	132
6.2	Future Work	135
6.2.1	Leveraging Topic Models for Biclustering	135
6.2.2	Improving the Inference of Topic Models	137
	References	141
	Appendix A Inference of DT2B via Collapsed Gibbs Sampling	150
A.1	Posterior derivation given the conditions of the model	150
A.2	Posterior derivation for each record	154
A.3	Perplexity of a topic using DT2B	156

List of figures

2.1	Expanded graphical representation of the LDA model. The red arrows indicate the main independence assumptions about the LDA model. a) The order of documents is irrelevant b) The order of words is irrelevant c) Topic dependency is only given by a Dirichlet prior distribution parameterized by α d) The number of topics must be given a priori. . . .	13
2.2	Bicluster structures. Figures modified from [70]	20
2.3	A simple topic model and an approximate topic model assuming some independence between the parameters in the model	26
3.1	Graphical representation of the categorical mixture model.	32
3.2	Graphical representation of the latent Dirichlet allocation model.	33
4.1	Information extracted by an LDA (left) and a Dual LDA (right). The width of the arrow represents the preference of a topic. The ovals represents the topics.	54
4.2	Information extracted by DT2B. The ovals represents the topics and the width of the arrows represents the strength of the relationship between two topics.	55
4.3	Implementation of intuitions 1 and 2.	56

4.4	Function used to define the <i>Categorical</i> ² distribution. The visualization indicates the mapping from the parameters to the support set.	58
4.5	Dual topics to bicluster graphical model	59
4.6	Demonstration of how DT2B extracts bicliques.	67
4.7	All influences of all pairs in the dataset.	68
4.8	Recovery and relevance results for bicluster classification. The datasets are visualized with a matrix whose color represent the number of visits of users (rows) to places (columns).	72
4.9	Different initialization methods for extracting biclusters with DT2B . .	73
4.10	Proportion of the enriched biclusters for DT2B. The results are compared with the biclustering algorithms studied by Eren et al on five different significance level α [32].	74
4.11	Reviewer recommendation	76
4.12	Comparison for place-topic extraction between LDA and DT2B. DT2B was initialized with the results of the LDA. The numbers represent the number of samples allocated to a topic	79
4.13	Results of Table 4.5 for every user, we sort the users and places by their average Battacharyya distance to facilitate the visualization of the results.	81
4.14	On top, we show four movie-topics represented by a) Most representative movies and its year of release. b) Histogram of the movie genres weighted by the number of instances that a movie was allocated to a topic. On the middle we represent the correlation between topics using a parallel sets visualization. On the bottom, we show the user-topics represented by a) Histogram of the user's profession b) Histogram of the user's age. Both histograms are weighted by the number of instances that a user was allocated to a topic.	83

4.15	Author-topics, word-topics and its interrelation. The width of the box surrounding each topic represents the total amount of samples allocated to the topic.	85
5.1	Using the function $g()$ to transform an r.v. Z_j with categorical distribution to an r.v. Y_j with degenerate Dirichlet distribution.	90
5.2	Approximation of the posterior $p(Z_j Z_{-j} = z_{-j}, \mathbb{D})$ via sampling. The figure shows examples of m draws from the posterior. As the number of samples grows, we obtain better estimate of the distribution. The samples approach the posterior itself when m grows up to infinity. . .	101
5.3	Behavior of $p(U)$ as $m \rightarrow \infty$ for $\lambda_k = 0.3$	104
5.4	Connection between the inferential methods.	111
5.5	Predictive perplexity on the GPU based inference methods versus the number of document-word pairs per batch.	116
5.6	Predictive perplexity on the GPU based inference methods versus the number of document-word pairs per batch.	117
5.7	Average running time per iteration versus the number of batches. . . .	122
5.8	Average running time per iteration versus the number of batches. . . .	123
5.9	Predictive perplexity of DT2B inferred by Heron or CGS for various values of Kr and Kc on the Lastfm dataset.	125
5.10	Predictive perplexity of DT2B inferred by Heron or CGS for various values of Kr and Kc on the NIPS dataset.	125
5.11	Predictive perplexity of DT2B inferred by Heron or CGS for various values of Kr and Kc on the Movielens dataset.	125
6.1	The plate diagram for PTM. Figure from Harvey et al. [50]	136
6.2	PTM model including the interrelation between user-topics and word-topics.	136

List of tables

2.1	Main assumptions of LDA and the label for the family of models that have proposed a solution for relaxing the assumption	14
2.2	Characteristics of Biclustering Algorithms	23
3.1	Comparison between clustering and biclustering in three different scientific fields.	35
3.2	Sample mobility dataset for 6 users and 7 places. The expected biclusters are presented in blue and red.	37
3.3	Posterior density as a function of the statistics Ω	44
4.1	Variable definitions for the dual topics to bicluster model.	60
4.2	Gene expression omnibus datasets.	70
4.3	Five most enriched terms on GDS589.	75
4.4	Average misclassification error for ten random initializations. The initialization method is specified in the headers of the table. The asterisk represents those cases where the models completely confused two topics.	77

4.5	On the left side of the table, we show the average Battacharyya distance between the approximated distributions $\hat{P}_{LDA}(P U)$, $\hat{P}_{DT2B}(P U)$ and the estimated distribution $P(P U)$. We assume the maximum likelihood estimation of $P(P U)$ as ground truth. On the right side, we display the Battacharyya distance between the approximated distributions $\hat{P}_{LDA}(U P)$, $\hat{P}_{DT2B}(U P)$ and the estimated distribution $P(U P)$. We assume the maximum likelihood estimation of $P(U P)$ as ground truth.	81
5.1	Hyperparameter setting per dataset.	116
5.2	Predictive perplexity versus the number of topics.	117
5.3	Predictive perplexity versus the hyperparameters α and β	119
5.4	Average running time per iteration versus the number of topics.	121
5.5	Automatic topic coherence of LDA inferred by CGS, SAME, and Heron. Best results per dataset use bold font. From top to bottom, the datasets are Cora, Diggs, and Movielens, respectively.	124
5.6	Hyperparameter values of sLDA as found for the classification of 20 news dataset using CGS, SAME, and Heron.	128
5.7	Hyperparameter values of sLDA as found for the classification of news articles using CGS, SAME, and Heron.	128
5.8	Micro-F1 score for the classification of newsgroups posts using sLDA as inferred by CGS, SAME, and Heron	129
5.9	Micro-F1 score for the classification of Reuters news articles using sLDA as inferred by CGS, SAME, and Heron	129

Chapter 1

Introduction

1.1 Background

Topic models are a type of generative statistical model for inferring semantic topics from text corpus. The essence of the topics is to group the words so that the documents in the corpus can be summarized with a mixture of the extracted topics [13].

The extracted topics have found applications in text and language processing. For example, Chang and Blei have shown how to incorporate document citations as a network of documents so that the topics found for new documents may reveal a set of related documents. This information is beneficial for the recommendation of documents [22]. Another example presented by Hai et al. is a topic model that simultaneously model aspect terms and opinion words for sentiment detection in user-generated reviews [48]. Besides text data, topic models have also been widely used in other fields. For instance, in computer vision, Cao and Fei-Fei showed that the topic assignments given to sub-regions of an image may be recombined into meaningful segments with high accuracy [17]. Other fields where topic models have been applied to may include music information retrieval [55] and computational biology [23].

Perhaps one of the most representative topic models would be the Latent Dirichlet Allocation model (LDA). LDA models every document as a mixture of topics, where each topic represents a distribution over the vocabulary of words. While LDA can successfully determine topics present on a corpus, the model makes some assumptions that constrain the learning of the topics. For example, LDA sets the number of discovered topics a priori. Also, the definition of the topic is not allowed to change over time. One branch of the study of topic models typically aims to relax the assumptions and limitations of LDA to give birth to novel applications.

Besides modeling assumptions, the learning of topic models has the main challenge of being intractable. The information of the posterior of topic models requires to compute a series of integrals whose analytic solution is unknown. Therefore, the inference of topic models relies on approximations with high computational cost and reduced predictive power. Most research in topic models focus on addressing assumptions of LDA as well as improving the inference of topic models. In this thesis, we aim to address one modeling assumption of LDA and we also aim to improve the inference of topic models.

1.2 Research Problems

Directionality of LDA. A particular issue of LDA is its directional approach to generating the data. The directionality of LDA stems from its hierarchical structure where at the top of the hierarchy LDA models the hyper-parameters that generate the documents, the second level uses the documents to generate its main topics, and at the last level, the LDA uses the obtained topics to generate the words that describe the topics. This hierarchical structure facilitates modeling of the probability of observing a word given a document, but it hinders the calculation of the probability of observing a document given a word. This is a disadvantage because the latter

probability represents useful information for applications in document recommendation and information retrieval. Furthermore, the topics of LDA can be used for clustering the words, but the generative process of LDA does not cluster the documents. A second LDA may model the opposite hierarchy placing the words at the top and the documents at the bottom so that the topics of the second LDA represents clusters of documents, but then we sacrifice the information of the topics composed by the words. We refer to the second LDA as the dual LDA because the second LDA is an alternative LDA model that can fit the same dataset to obtain complementary information. While LDA models the probability of a document given a word and obtains groups of words known as topics, the dual LDA models the probability of a word given a document and obtains groups of documents that we may label as document topics. The directionality of LDA is an issue because applications such as document recommendation may benefit from mining both types of clusters using one unified probabilistic model.

Improving the trade-off between computational efficiency and predictive power. To deal with the intractability of inference of topic models, researchers have found that a collapsing technique combined with Gibbs sampling (CGS) can be used to infer the parameters of some topic models. However, the scalability of this algorithm to larger datasets is compromised because the sampling methods are inherently sequential and hence, they are not prone to be implemented in a distributed fashion. The most common alternative to collapsed Gibbs sampling is variational inference (VI) [11]. VI proposes an optimization-based method that approximates the posterior of the model using a simplified version of the model with additional independence assumptions. As a main advantage over CGS, VI's optimization problem can be set up so that the learning can be distributable. As a main disadvantage, there exists a bias error introduced by the fact that the simplified model cannot cope with all the dependencies of the model that it approximates. The bias problem of VI translates in a reduced predictive

power of the model that may affect the performance on some applications. Numerous approaches have been studied to improve the running speed of the collapsed Gibbs sampler as well as VI's predictive power. Typical approaches include a combination of both methods. In most cases, there exists a trade-off between efficiency and predictive power that does not allow to improve the Pareto optimality of the learning of topic models [107].

1.3 Proposed Solutions

Using dual topic models for mining biclusters. In this thesis, we study how to extend LDA so that it can incorporate the information given by the probability of observing a document given the set of words. Such modeling empowers topic models to cluster two variables simultaneously. For instance, if we apply LDA on mobility data, we may choose to set the users at the document level and the places at the word level. LDA would therefore extract clusters of locations. But, if instead we set the locations at the document level and we set the users at the word level. Then, we would obtain clusters of users. Intuitively, by combining both modeling approaches we could simultaneously extract clusters of users and clusters of places and reveal the information of “Which groups of users visit which types of locations” . The simultaneous clustering of two variables is known as biclustering.

In Chapter 4, we propose to combine the two directional approaches of LDA into a single unified model obtaining topics that group the words and topics that group the documents. By simultaneously clustering words and documents, our proposed model is able to mine biclusters. We label our model as Dual Topics to Bicluster (DT2B). DT2B captures the interrelation between the two types of topics. We show that our proposed method improves the state of the art when mining biclusters in microarray and location data. In addition, we compare DT2B with other well-established topic models using a

reviewer recommendation based on text data. Finally, we show a theoretical connection that proves that DT2B generalizes LDA as well as the author-topic-model [87].

Improving the inference of topic models. Motivated by the need for improving the learning and efficacy of our model, in Chapter 5, we study the collapsed Gibbs sampler down to its roots in the Markov Chain. We find that Gibbs sampling is using a randomized approach that sacrifices of the assessment of convergence. We show that as an alternative, we can improve the learning of the parameters of the model if we apply a transformation, specifically, instead of assigning one topic to each word, the learning allows assigning a mixture of topics to each word. The transformation not only allows LDA to have a more flexible estimation of the parameters, but it deals with the assessment of convergence. This ultimately results in an improved predictive performance as well as reduced sensitivity to the hyperparameter settings. As a theoretical connection, we find that applying variational inference to our transformed model is equivalent to applying Gibbs sampling. To the best of our knowledge, this is the first time that a connection between the most common inference methods for topic models is found. We label our method as recursive compound allocation (RCA).

We continue our study regarding the efficiency and distributability of Gibbs-based inference methods and find that the equations that drive the learning of topic models are ultimately solving a nonlinear system of equations. This is beneficial, because there exist distributable methods that solve non-linear system of equations. We coin this method as Heron inference (Heron). We implement Heron using one GPU and show that Heron has significantly faster running time compared to the state augmentation for marginal estimation inference approach (SAME) [117], the most efficient state of the art method within the Gibbs framework. We also present theoretical connections between SAME and our inference method, which reveal why our method also improves the predictive power and efficiency of SAME.

For the empirical evaluation, we measured the accuracy of these models on the word prediction task which is perhaps the most common way to evaluate the predictive power of a topic model [22], [60], [46]. The word prediction task corresponds to the query: ‘*Given a new document, what is the likelihood of its words?*’. This task is commonly referred to as the predictive perplexity. In addition, we studied the behavior of the predictive power as we learn the parameters using more cores. While the state of the art inference algorithms sacrifice some of the predictive power to increase their scalability, we note that Heron is able to maintain the predictive power up to a large number of cores.

In addition, we also evaluated the quality of the extracted topics in terms of topic coherence [63]. We found that Heron is able to improve on the topic coherence when compared with SAME and the standard Gibbs sampler algorithm [117]. The standard evaluation metrics show the potential of Heron inference to improve real world applications solved by topic models. To illustrate one example, we also applied Heron inference to document classification using two news datasets.

1.4 Contributions of this Thesis

The thesis has made the following significant contributions:

1. We define the bivariate categorical distribution as a probability distribution for modeling the dependency of a pair of latent random variables. Using this distribution, we construct a novel unified probabilistic model to mine biclusters on bivariate categorical data. We label the model as Dual-Topics to Bicluster (DT2B). In addition, we prove that DT2B generalizes the author-topic model and the latent Dirichlet allocation model.

2. We develop a new inference method named recursively compound allocation (RCA) to estimate the posterior for latent topic models. In RCA, we propose to replace each topic label with a random variable representing all possible topic labels with their respective probabilities. The modification enables the models to increase their predictive power. Another key benefit of RCA is that it is deterministic compared to the stochastic nature of CGS. Determinism facilitates the assessment of convergence which in the case of other Gibbs-based methods requires additional heuristic measurements. To the best of our knowledge, RCA is the first deterministic learning algorithm within the Gibbs framework.
3. We study the convergence of RCA and find that RCA is ultimately attempting to solve a non-linear system of equations. The system of equations representation of a topic model allows us to solve the inference of topic models using a novel iterative method which we label as Heron Inference. We demonstrate theoretical support for distributability of Heron Inference and an optimization for avoiding computations of repeated tuples. In addition, We show how to derive Heron inference for a family of probabilistic models.
4. We present theoretical connections between RCA, Heron, SAME and variational inference. The connection between SAME and RCA serves to explain why RCA improves the predictive power of SAME, while the connection between Heron and RCA helps to explain why the algorithms converge to the same result. Finally, the connection between variational inference and RCA presents a bridge between optimization and Bayesian approaches to inference. This connection may be exploited to bring additional benefits of Bayesian approaches into variational inference or vice versa.

1.5 Summary

In this chapter, we briefly introduced topic models and some of the fields where topic models have found applications. Then, we presented main limitations and assumptions of topic models from where we chose the two particular research problems that we address on this thesis: The directionality of LDA and the trade-off between computational efficiency and predictive power of existing inference algorithms. We explained the motivation for resolving the research problems and we introduced the proposed solutions that we further explain in the subsequent chapters of the thesis. Finally, we explicitly mention the contributions of this thesis.

Chapter 2

Literature Review

Topic models have gained popularity because of their ability to extract the topics that summarize the documents in a corpus. The model fits the corpus and extract the topics as groups of words. Since topic models are a generative model, new documents can be sampled from a fitted model and therefore they are considered to be a language model. Notably, topic models make several assumptions that allows them to extract the main topics in a corpus without being able to generate coherent documents. The extracted topics have a high level of coherence with the human concept of a topic and it has been argued that relaxing many of the assumptions of the model may improve the quality of the extracted topics in terms of coherence [77] and predictive power [9]. For example, Latent Dirichlet Allocation (LDA), one of the most representative topic models, does not model the possible correlation between the topics. Naturally, A topic related to politics may be more related to a topic related to economics than to a topic about chemistry. Another assumption is given by the fact that the topics do not evolve in time which may not hold for well some datasets. For example, the topics of a twitter dataset will likely change in time because daily events trigger different news constantly. We will explore in Section 2.1 more examples of assumptions made by LDA and possible solutions presented in the literature.

Topic models have also been employed to fit other types of data such as microarray data for exploring biological phenomena and also location data to summarize the mobility patterns of a group of users. In the latter domain, we apply LDA to the data by setting the user at the document level and the places at the word level. Under this setting, the document represents the locations visited by an individual and the model is able to extract the groups of places that a population visits. Alternatively, if we set the places at the document level and the locations at the word level, LDA would extract the groups of users that visited a location. If we had a unified topic model that could simultaneously extract both groups of users and groups of places, we could mine the information about “which group of users visit which groups of places”. For example, the model may extract that a group of users, which roughly correspond to students, typically visit a set of locations near a campus. This information may be desirable for recommendation tasks or urban planning. The process of simultaneously grouping two different variables is known as biclustering and it has been studied in the literature. However, a topic modeling approach to biclustering could improve several constraints of biclustering algorithms. In Section 2.2, we explain in detail the main algorithms proposed for mining biclusters and in Chapter 4, we present a new unified topic modeling approach to biclustering.

Another research problem within topic models is associated with the learning algorithms. There are two main problems that the literature attempts to address:

- 1. Accuracy and efficiency of learning algorithms.** A first issue with existent inference algorithms is that there exists a trade-off between predictive power and computational efficiency. On one hand, we have sampling algorithms which estimate with high precision the parameters of the model. However, these algorithms are sequential by nature and there is not trivial way to distribute their computation across several cores. Some algorithms have been proposed to handle this problem at the

expense of sacrificing the predictive power. On the other hand, we have optimization approaches which simplify the model by making additional independence assumptions. As a consequence, the optimization methods approximate the original model and sacrifice some of their predictive power. We present in more detail different approaches to the inference of topic models in Section 2.3.1.

With the exponential growth of data, it is imperative to find novel ways of learning the parameters of these models improving both the predictive power as well as the computational efficiency. Better inference methods speeds up research and development of novel topic models and applications. In Chapter 5, we present an inference method with these characteristics.

2. Automatic training of topic models. While the learning process of neural networks has reached a stage where a computer automatically learns the parameters of a given network architecture and a training dataset, in probabilistic models the learning algorithms rely on integrals that cannot be automated and require complex mathematical derivations and clever approximations to be able to learn new models. As a result, most research in machine learning has shifted from probabilistic approaches into deep learning approaches. However, despite the success of deep learning approaches, probabilistic models are arguably more comprehensible than neural networks. The parameters of probabilistic models represent probabilities that explain what the model has learned in terms of the given data. Some proposals have been made to learn probabilistic models automatically [85], [59]. These algorithms present a first step towards an automated learning algorithm that works well for a large class of models. However, some shortcomings need to be addressed before the automated learning can be applied to novel topic models. The automated learning algorithms use approximations to the original models that may compromise the accuracy of the learned model. Therefore, “Fake data” checks are advised to validate the learning of the models [26].

Also, the selection of an approximation model is a key design decision that may lead to a high-variance estimators of the objective function. Finally, it is required to develop an adaptive step-size optimization sequence that works well in both theory and practice. The problems of current automated learning algorithms motivate the study of novel learning approaches that may facilitate automatic learning of topic models. In Section 2.3.2, we present related learning algorithms that aim to solve a family of probabilistic models.

In Chapter 5, we present an inference algorithm that may help achieve a better understanding on how to approach automatic learning by narrowing a link between the learning of topic models and the learning of neural networks. We present the details of such connection in Section 6.2.2,

2.1 Topic Models and its Applications

Topic models are commonly used to organize documents by using the set of words written in them [95, 61]. However, topic models have also been widely used in computer vision [17], text data mining [22], and [73], natural language processing [48], music information retrieval [55], and computational biology [23].

One of the most typical topic models would be the LDA [13]. LDA is able to discover K topics that pervade an unstructured collection of documents. In LDA, we assume that each document has a multinomial distribution over K topics parameterized by θ . We use θ_d as the parameter for representing $p(Z|d)$, the probability distribution of the topics in a document d . Each topic is defined as a multinomial distribution over the vocabulary of words parameterized by ϕ . We use ϕ_w as the parameter for representing $p(w|Z)$, the probability of observing the word w given a topic. The per document topic distribution θ_d is generated by sampling from a Dirichlet distribution

Fig. 2.1 Expanded graphical representation of the LDA model. The red arrows indicate the main independence assumptions about the LDA model. a) The order of documents is irrelevant b) The order of words is irrelevant c) Topic dependency is only given by a Dirichlet prior distribution parameterized by α d) The number of topics must be given a priori.

with hyperparameters α , and the per topic word distribution ϕ_w is generated by sampling from a Dirichlet distribution with hyperparameters β .

The model has made major assumptions which hold well when the task is to extract the main topics. However, some assumptions have been released with the intention to use topic models in other applications or to improve the modeling of some datasets. The major assumptions of LDA are shown using an extended graphical model in Figure 2.1. We review in particular four main assumptions: a) The order of the documents is not important. The model clearly assumes that all documents are drawn from a Dirichlet distributions where the ordering of sampling is not considered b) The order of the words is not important. Similarly to the documents, the words are also drawn from a distribution where the ordering of the appearing words cannot be enforced c) The topics are independent. As presented by Blei et al., the topics are independent because they are drawn from a Dirichlet distribution which can always be approximated using

LDA assumption	Family of models
Order of documents is not important	N-gram topic models
Order of words is not important	Temporal topic models
The topic correlation is not modeled	Correlated topic models
The topics must be known a priori	Bayesian non-parametric topic models

Table 2.1 Main assumptions of LDA and the label for the family of models that have proposed a solution for relaxing the assumption

several gamma distributions, one for each topic [60] d) The number of topics K is a hyperparameter of the model and it must be appropriately chosen for each application. Table 2.1 presents the assumptions made by LDA and the common name for the family of models that presents a solution to eliminate the assumption.

Next, we will review how these families of models relax the main assumptions of LDA. Then, in Section 2.1.5, we will review how researchers have extended LDA by incorporating the modeling of more data in addition to the intended document-words pairs that LDA uses in its architecture.

2.1.1 N-gram topic models

LDA is a bag of words model. This assumption implies that the ordering of the words inside a document is irrelevant [13]. Some researchers have addressed this problem by conditioning the generation of a word on the previous n -words. Hanna Wallach proposed a bigram topic model, where topics would be extracted as pairs of words [101]. This idea was extended by the distant n -gram topic model [105], where the authors focused in extending the bigram models by considering the dependencies between a word and the n -th previous words. The n -gram topic model tends to create an exponential number of parameters as n increases. One possible solution is to use a hidden Markov model (HMM) to model the dependency between words. This idea was developed by Griffiths et al, their composite model switches between LDA and a standard HMM so that the dependencies between words are governed by the HMM

and the content of the topics is governed by the LDA [46]. An alternative modeling relaxing the bag of words assumption is the hidden topic Markov model (HTMM) [47]. HTMM considers the relationship between the topic assigned to the current word and the topics assigned to several previous words instead of considering the relationship between the current word and the previous word.

2.1.2 Bayesian non-parametric topic models

LDA assumes that the number of topics must be known a priori. This assumption has been relaxed with the introduction of the hierarchical Dirichlet process (HDP) that allows the extraction of the number of topics of an LDA model without a explicit specification [16]. HDP operates by assuming that there exists a finite number of groups sharing the same set of mixture components with different mixing proportion. To describe the mixing proportions of each group, one probability measure is assigned to each of the groups using a Dirichlet process. The base distribution shared as a parameter among the Dirichlet processes cannot control the trade-off between the desired property of sharing clusters among groups and allowing all possible mixing combinations of clusters in a given group. As solution, Teh et al., considered using a prior over this base distribution [10]. Due to the benefits of using conjugate priors, the authors propose using another Dirichlet process as prior distribution, hence the name of the Hierarchical Dirichlet process [97]. The number of topics is modeled as infinite and the relevance of each topic is represented by sampling from Dirichlet process. One sample of the Dirichlet process will yield a discrete distribution whose support has finite dimensionality. Since every dimension represents a topic, the extracted number of topics will be finite as well. The Dirichlet process has several construction definitions being the stick-breaking construction the simplest known [57].

2.1.3 Correlated topic models

Another assumption of LDA is the near independence among the topics, the Dirichlet hyper prior used in LDA as prior knowledge about the topic distributions can always be replaced by one independent Gamma distribution per topic. The correlated topic model [60] has been designed to relax this assumption by changing the Dirichlet hyper-prior for a multivariate log-normal hyper-prior. The multivariate log-normal distribution models the correlation between the topics by using its parametrization given by a covariance matrix and a mean vector. Every parameter of the covariance matrix is used to encode the correlation between each pair of topics, while the mean vector represents the importance of the topics.

As an alternative for modeling the correlation between topics, Li et al. proposed the pachinko allocation topic model (PAM) which uses a directed acyclic graph to represent the topics with each non-leaf node and the words with the leaf nodes [66]. Hence, the relationship between all nodes represents the distributions of topics over topics and distributions of topics over words. PAM is a generalization of LDA since a graph where all nodes are connected to only one node will correspond to the standard LDA. The correlation between topics is given by the hierarchical relations existent between topics. PAM has an advantage over the correlated topic model given by the possibility of a sparse parameterization of the topic intra-correlation. In the correlated topic models the correlation between all topics must be parameterized.

2.1.4 Temporal dynamics of the topic models

A final assumption of LDA is the exchangeability of the documents. In other words, the ordering of documents is irrelevant when doing the inference of the topics and the topics will not exhibit changes in time. A set of topic models represent the temporal dynamics of the data by modeling the latent topics as changing throughout time.

Blei and Lafferty presented the dynamic topic model (DTM) that uses a Kalman filter to model the temporal alignment among topics across collections of documents [12]. As a consequence, DTM uses the Markov assumption that the future state of the topics only depends on the inference values of the present state of the topics. Several models have proposed different ways to improve DTM. For instance, Wang et al. proposed the Dynamic Mixture Model(DMM) [106] to force temporal dependence between documents rather than the temporal dependence between collections of documents, this allows a finer modeling of the evolution of the per document topic distributions. Similar approaches are the Sequential LDA (SeqLDA) [31] and the evolutionary HDP (EvoHDP) [115]. Both models propose to change the Dirichlet prior of DTM. SeqLDA uses a Pitman-Yor process to improve the space and computational complexity of the inference process, while EvoHDP uses a Hierarchical Dirichlet process to automatically determine the number of topics including emergence and disappearance.

Also, in topics over time model (TOT) [104], the authors relaxed the Markov assumption of DTM by treating time as an observed continuous variable such as the topics. This helps to avoid a Markov model's risk of inappropriately dividing a topic in two when there is a brief gap in its appearance. However, because all the topics are drawn from one distribution, TOT and DMM are not the most appropriate models for identifying topics that change in time.

The dynamic topic model remains as the only model that allows topics to be generated from different distribution and it also models the change of topics over time. However DTM, also known as the discrete time dynamic topic model (dDTM) splits the data into groups based on temporal discretization and thus, it requires to choose a discretization value. Because of computational limitations, the discretization value is chosen based on the trade off between accuracy and computational complexity. To solve this issue, Wong and Blei proposed the continuous time dynamic topic model

(cDTM) which corresponds to the infinitesimal discretization of the dDTM. cDTM uses a Brownian motion to model the evolution of the topics in time [103].

2.1.5 Extending LDA with additional variables

Aside from solving the assumptions of LDA, some researchers have studied extending an LDA model by including additional variables in order to better fit particular supervised learning applications such place recommendation or next place prediction. For geographic-textual data, researchers have studied different variables as users, geographical and functional region, time and text [111, 6, 74, 113]. For check-in data researchers have focused on users, time, item preference, geographical and functional regions [67]. All the mentioned models represent a user as a document, hence we recognize these models as personalized approaches.

Other extensions of the LDA model have allowed the exploration of non-personalized models. Non-personalized models are important, as they provide essential information in recommendation tasks [14]. The author topic model (ATM) is an extension of LDA where each word in a document is assumed to be written by a specific user, and a document can be written by several authors [87]. This model has been used in the human mobility behavior field by modeling the document as time of the day and the authors as the users. As a result, ATM model can extract place-topics of interest for a set of users in a particular time of the day [35]. The author topic model has also been used for aggregating locations instead of users in the place-recommendation task [68].

2.2 Biclustering

In this section, we review major biclustering algorithms available in the literature. We can categorize the algorithms based on the structure and type of the biclusters extracted [70].

2.2.1 Biclustering

Biclustering algorithms have been used to cluster simultaneously two variables [49]. While it is understandable that clustering one variable is beneficial for organizing or summarizing information, it is not intuitive why do we need to extract biclusters. To see why extracting biclusters can be useful, let us consider a simple example:

Tom is planning a party for his new four-room flat. Each room has a separate sound system, so he wants to play a movie in each room. As a host, Bob wants everyone to enjoy the movie. Therefore, he needs to distribute movies and guests to each room in order to ensure that each guest watch their favorite movie. Tom has invited forty guests, and he owns twenty movies. He sends out a survey to each guest asking to rate from 1 to 5 each movie. After receiving the guest's responses, Tom collects the data into a 40×20 matrix A , where $A_{i,j}$ stores the ranking of j -th movie given by the i -th guest.

A biclustering algorithm would take as an input the matrix A and it would output a list of biclusters where each bicluster is defined by a subset of the guests and a subset of the movies. After biclustering, the rows and columns of the data matrix may be reordered to show the assignment of guests and movies to the rooms in the flat.

Biclustering can be applied to any pair of variables whose type of relationship is described as *many-to-many*. For example, users and movies, documents and words, or users and webpages.

Fig. 2.2 Bicluster structures. Figures modified from [70]

2.2.2 Biclustering structures

The Biclustering structure represents the shape of the biclusters that can be extracted from a matrix. The different bicluster structures are shown in Figure 2.2. The description is given below :

Exclusive row and column Every user and every place is assigned to exactly one bicluster.

Disjoint Biclusters Every user and every place is assigned to at most one bicluster.

Exclusive row biclusters Every user is assigned to one bicluster, but every place is assigned to at least one bicluster.

Exclusive columns biclusters Every user is assigned to at least one bicluster, but every place is assigned to one bicluster.

Checkerboard structure Every user-place pair is assigned to exactly one bicluster.

2.2.3 Biclustering types

The biclustering type refers to the relationship in the matrix that forces the rows and the columns to form a bicluster. There are two main types of biclusters: Biclusters with constant values and biclusters with coherent values. The first type corresponds to

those biclusters whose elements have the same or a similar value. There exist several subtypes, e.g., biclusters with same values on the rows or biclusters with same values on its columns.

The second type corresponds to those biclusters whose elements have some coherence. For example, either the rows, or the columns are dependent according to some mathematical condition or the values in the bicluster follow a known distribution. In particular, when the coherence is defined to extract biclusters whose rows and columns have a high-frequency of occurring in the dataset, the biclusters obtained corresponds to the bicliques of the dataset.

2.2.4 Existing biclustering algorithms

In this section, we briefly review related biclustering algorithms that best performed on tested data with varying conditions, including varying noise and varying numbers of biclusters [32, 70].

Cheng and Church

Cheng and Church proposed the first biclustering algorithm [24]. Their algorithm is famous for having the theoretical guarantee of finding the bicluster where its rows and columns are shifted versions of each other. i.e., the bicluster is represented by a matrix with rank equals to one. Their algorithm is the most common baseline among biclustering algorithms [70]. The algorithm operates by assuming that all the data corresponds to a bicluster, and then iteratively it removes the row or the column with lowest similarity score to the rest of the bicluster. At each iteration, the algorithm evaluates the candidate bicluster using a function score and when the bicluster reaches its smallest size (last iteration); it retrieves the bicluster with highest score. If a second bicluster needs to be found, the data from the first algorithm is replaced by noisy

data and the algorithm is restarted. As one limitation, Cheng and Church biclustering algorithm extracts a bicluster at a time.

xMOTIF

xMOTIF [76] aims to find a subset of rows that occur simultaneously across the columns under a set of conditions. E.g., a linear order across the columns. xMOTIF retrieves The largest bicluster that contains the maximum number of conserved rows. Similarly to Cheng and Church algorithm, xMOTIF extracts one bicluster at a time.

QUBIC

While the methods explained above identify patterns under parameterized conditions such a linear scaling among the rows or columns. Quantitative Biclustering (QUBIC) attempts to find similar patterns based on positive and negative correlation, so that the algorithm is less sensitive to outliers in the data [69]. The algorithm uses a combinatorial technique that has demonstrated an improvement over other combinatorial techniques such SAMBA [96], ISA [56], Bimax [83]. The discovery of biclusters follows the same procedure as [24] and [76], that is, biclusters are discovered one at a time.

Spectral Bipartite Graph

Table 2.2 presents the main characteristics of existing biclustering algorithms. We note that their main limitation is that the biclustering algorithms assume a bicluster structure. We can use a probabilistic modeling approach based on topic models to achieve a biclustering algorithm free from any structure constraint. We present this model in Chapter 4.

Table 2.2 Characteristics of Biclustering Algorithms

Algorithm	Type	Structure	Discovery	Approach
Block Clustering	Constant	a	One at a time	Divide and Conquer
Bimax [83]	Coherent Values	b	One at a time	Greedy
FLOC [71]	Coherent Values	a,b	Simultaneous	Greedy
PLAID [65]	Coherent Values	a,b	One at a time	Probabilistic
CTWC [20]	Coherent Values	b	One at a time	Combinatorial
ITWC [41]	Coherent Values	b	One at a time	Combinatorial
Spectral [28]	Coherent Values	a	Simultaneous	Optimization
QUBIC [69]	Coherent Values	a,b,c	One at a time	Optimization
xMotif [76]	Coherent Values	a,b,c	Simultaneous	Greedy
SAMBA [96]	Coherent Evolution	a,b,c	One at a time	Combinatorial
Bayesian [92]	Coherent Values	e	Simultaneous	Probabilistic
Cheng [24]	Constant	a,b,c	One at a time	Iterative
Mondrian [88]	Coherent Values	e	Simultaneous	Probabilistic

2.3 Inference of Topic Models

Designing topic models requires more than understanding the relationships between the problem variables and relaxing assumptions. The computational complexity of LDA-based topic models mentioned in Section 2.3.1 has been proved to be NP-hard [94]. Additionally, existent inference algorithms have different advantages and there is no solution on how to generalize the inference process by keeping all advantages. As a consequence, deep understanding of inference methods is necessary in order to design and apply new topic models to large datasets. In this section, we review the approximation algorithms that have been proposed in the literature to solve the intractability problem of topic models. We classify these algorithms in two categories: the model-based inference methods and the family-based inference methods.

2.3.1 Model-based inference methods

Model-based methods correspond to those techniques that must be tailored for each particular topic model. To solve the intractability of topic models, Hoffman et al.

proposed to use an expectation-maximization algorithm (EM) [54] for an LDA model with no priors, also known as probabilistic Latent Semantic Analysis. The algorithm aims to find the corresponding values of the latent variables and the given parameters of the model based on the conditional dependencies implied by the model. The same approach works for LDA using Dirichlet priors [75]. A pragmatic improvement over EM has been achieved by using the variational EM framework (VI) [13] in which a new simplified tractable model is used to find the parameters of the intractable model. The simplified model makes the assumption that the latent variables and the parameters are mutually independent. Rather than solving the simplified model, VI method solves the simplified model whose Kullback-Leibler Divergence to the original model is minimized. This problem is equivalent to the problem of maximizing the free variational energy which is a known framework for solving the inference of probabilistic models. The equivalence between the problems can be shown using Jensen's Inequality [13] or by manipulation of the equations [36].

From a different perspective, Griffiths and Steyvers proposed a Markov chain monte carlo approach based on collapsing or integrating out the parameters of the model which is analytically feasible since the prior distributions are conjugate to the distributions assumed by the model, the model was labeled as collapsed Gibbs sampling (CGS) [45]. The advantage of CGS over VI is the existence of theoretical bounds over the convergence of the inference process, however CGS requires conjugate priors in order to collapse the parameters of the model. Teh et al. proposed Collapsed Variational Bayesian Inference (CVB) [98]. CVB is a variational Bayesian algorithm that uses the collapsed Gibbs sampling approach of collapsing the parameters by integrating them out from the joint probability distribution of the model. So instead of approximating the model with a simplified version of the model, CVB approximates the collapsed version of the model using variational inference. As a result, CVB is able to relax some

of the the independence assumptions of VI. Although, the latent variables are still assumed to be mutually independent given the words and Θ .

2.3.2 Family-based inference methods

More recent algorithms aim to develop inference techniques for a family of probabilistic models. Blei et al. proposed stochastic variational inference (sVI)[53], an algorithm to approximate the posterior distribution using subsets of the data rather than the entire dataset. Their method works for a family of models with i) Conjugate priors ii) The generative process follows a conditional chain. E.g., in LDA, Θ is conditioned by α , Φ is conditioned by β and w is conditioned by Φ_z iii) the distributions assigned to the random variables must belong to the exponential family. Similarly to Gibbs sampling, sVI can be applied to any model whose complete conditional distribution is the set of conditional distributions of a hidden variable in terms of all other hidden and non-hidden variables in the model.

Using the conditions i, ii, and iii, sVI is able to transform any model in terms of the simplified topic model presented in Figure 2.3a. The simple topic model p is composed of a local latent variable z linked to every instance of the data x and global latent variables φ governing all instances of the local latent variables. Based on this Bayesian formulation, β corresponds to the given hyperparameters of φ . Solving the simple model will be enough to solve a family of topic models. The authors of this work have decided to solve the inference of the simple topic model using variational inference. The approximate family of models q is given in figure 2.3b. Ideally, we want to select the model $\hat{q}$ that is more similar to the original model given by p , that is $\hat{q} = \underset{q}{\operatorname{argmin}} KL(q(z, \varphi) || p(z, \varphi | x))$. Using Jensen's inequality, the authors show how this problem is equivalent to maximizing the evidence lower bound (ELBO). i.e., maximizing the

(a) A simple topic model with a local variable z and a global variable φ (b) Approximate variational model of the simple topic model

Fig. 2.3 A simple topic model and an approximate topic model assuming some independence between the parameters in the model

right hand side of equation 2.1

$$\log p(x) \geq E_q[\log p(x, z, \varphi)] - E_q[\log q(z, \varphi)] = L(q) \quad (2.1)$$

Because Equation 2.1 depends on p , solving the inference problem will depend on p having conjugate priors and using distributions in the exponential family (conditions i and iii). This model can be applied to models such as Bayesian mixture models [42], hidden Markov models [43, 38], Kalman filters [58, 37], probabilistic factor analysis/matrix factorization models [79, 25] and some Bayesian non-parametric mixture models [33, 98].

More recently, black box variational Inference [85] was introduced as a generalization of stochastic variational inference by removing conditions i) and iii). The relaxation is possible since the objective function $L(q)$ is solved using stochastic optimization and Monte Carlo estimation [78]. Stochastic optimization is used to find the maximum value of $L(q)$ using noisy estimates of $\nabla_{\varphi} L$ in a framework provided by Robbins-Monro [86]. The framework guarantees convergence if the following iterative process is satisfied:

$$q(\varphi)_{t+1} \leftarrow q(\varphi)_t + \rho h_t(q) \quad (2.2)$$

where ρ corresponds to the learning rate and $h_t(q)$ is a distribution whose expectation approximates $\nabla_\varphi L$. The authors use Monte-Carlo estimation to approximate $\nabla_\varphi L$ as follows $h_t(q) = \frac{1}{S} \sum_{s=1}^S \nabla_\varphi \log q(z_s|\varphi)(\log p(x, z_s) - \log q(z_s|\varphi))$ where z_s corresponds to a sample from the variational distribution as given by $z_s \sim q(z|\varphi)$.

2.4 Summary

In this chapter, we presented the literature relevant to the research problems that we address on this thesis. In particular, we presented the work relevant to three research fields: topic models, biclustering and the inference of topic models. We identified four main limitations of the influential latent Dirichlet allocation model and we categorized the relevant literature by presenting how the proposed methods address the presented limitations. With respect to biclustering, we grouped existing algorithms based on how the algorithm extracts a particular bicluster type and a particular bicluster structure. Finally, we categorized the inference of topic into “Model-based inference methods” for the inference algorithms that must be carefully fitted to each probabilistic model, and “family-based inference methods” for those algorithms that can be applied to a family of probabilistic models with minor adjustments to the algorithm.

Chapter 3

Preliminaries

In this chapter, we introduce the theory necessary to present the contributions of this thesis in the subsequent chapters. We will use two sections to present the work relevant to satisfy the two objectives of this thesis.

Our first objective is to extend LDA so that we can mine the relationship between two types of clusters, also known as biclusters. Such information may be used to mine different type of information when applied to different datasets. For example, using location data, we could extract the information that corresponds to "Which groups of users visit which types of locations". This information may be used to improve applications such as friend recommendation, location recommendation or simply to study the visit patterns of social groups. Using text data, we may extract the information about "Which authors write about similar topics". This information can be used to mine which authors have an interest or expertise in some domain.

The problem with LDA is that the model may identify the relationships between the words represented as the topics of a corpus, but it will not identify any relationship between the authors. In order to mine biclusters, it is necessary to extend the LDA model so that in addition to the groups of words, we can also extract groups of authors. To construct a model with these characteristics, we use Section 3.1 to dissect LDA

and understand how it groups the words and how we could use this idea to group the authors. In particular, we use Sections 3.1.1 and 3.1.2 to study the theory necessary to present the categorical mixture model (CMM). CMM is the model that empowers the clustering of words in LDA. Then, in Section 3.1.3, we study with greater detail LDA and how LDA can be used for biclustering. Finally, given that our aim is to extend LDA to extract biclusters, we present formally the biclustering problem in Section 3.1.4.

The other main objective of this thesis is related to the learning of topic models. Hence, we review the collapsed Gibbs sampling method applied to LDA in Section 3.2. We study the details of how Gibbs sampling is ultimately running a Markov chain whose target distribution corresponds to the fitted parameters of a probabilistic model. In addition, the Markov chain is used to explain the convergence of Gibbs sampling in Section 3.2.1. An in-depth understanding of Gibbs sampling is required to understand how we can improve the inference of topic models in terms of efficiency and predictive power. Hence, we present the theory of the collapsed Gibbs sampling algorithm in Section 3.2.2 and then we present how we can apply this theory to the inference of LDA in Section 3.2.3

3.1 Probabilistic Bayesian Models

Probabilistic models describe how a particular dataset is generated. The data assumes an underlying probability distribution that describes the probability of observing each of the instances in the dataset. As a simple example, we may have a dataset composed of a set of words. Using the probabilistic framework, we may assume that the words W are a random variable that follows a categorical distribution :

$$W \sim f(W|\phi_1, \dots, \phi_V) \tag{3.1}$$

where V corresponds to the size of the vocabulary used. Hence, the j^{th} word in the dataset will have a probability ϕ_j of being observed. Using this model we can easily generate as many words as desired.

Bayesian probabilistic models are a more robust modeling approach. Under this approach, the parameters that describe the probability of each data instance are known to have some uncertainty and so, the uncertainty is measured by defining the parameters as random variables. Using our example, the model must define a second probability distribution used to draw the parameters $\phi_1, \dots, \phi_V$. This second distribution describes the probability of each combination of the parameters that may exist. In this case, a typical choice is the Dirichlet distribution because the Dirichlet distribution is able to draw parameters whose values are chosen between zero and one and their sum adds to one. The density of the Dirichlet distribution is expressed as follows:

$$f(\phi_1, \dots, \phi_V | \alpha) = \frac{\Gamma(\sum_1^V \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i \phi_i^{\alpha_i - 1} = \text{Dirichlet}(\alpha) \quad (3.2)$$

Combining Equation 3.1 with Equation 3.2, we obtain the generation of the data in a two step model:

1. Draw the parameters from $f(\phi_1, \dots, \phi_V | \alpha)$
2. Draw words from $f(W | \phi_1, \dots, \phi_V)$

The model presented in Equation 2 is known as the Dirichlet-Categorical model (DCM). In the next section, we show that topic models are fundamentally based on DCM.

3.1.1 Topic models

Topic models were first introduced by LDA, a Bayesian probabilistic model designed to extract the latent topics existent in a collection of documents. The topics are defined to be distributions over the vocabulary. This modeling representation of the topics is based on the categorical mixture model (CMM). CMM is a generalization of the Dirichlet-Categorical Model (DCM) presented in the previous section.

3.1.2 Categorical mixture model

While DCM assumes a set of parameters $\phi_1, \dots, \phi_V$ to describe the generation of the data, CMM assumes that there exists K possible distributions each with V parameters $\phi_{11}, \dots, \phi_{KV}$ that may generate the data. The words are drawn from mixing these K distributions, and the mixing proportions are given by another categorical distribution with parameters $\theta_1 \dots \theta_K$. As a result, CMM is a probabilistic model used for clustering categorical data because it summarizes the dataset with the extracted K distributions.

CMM models the generation of N observations w_i $i \in 1 \dots N$ assuming the following process:

Draw the global mixing distribution $\theta \sim \text{Dirichlet}(\alpha)$

For each sample $1 \dots N$:

Select one categorical distribution (cluster) $z_i \sim \text{Categorical}(\theta)$

Sample a word from the cluster distribution $w_i \sim \text{Categorical}(\phi_{z_i})$

The topics in this model correspond to the information extracted by the distributions parameterized by ϕ . The term *topic* originated when this model was applied to a set of words because in that scenario, the parameters ϕ represent distributions over the vocabulary of words, and this grouping of the words can be semantically thought of being a topic. Finally, we present CMM's graphical model in Figure 3.1.

Fig. 3.1 Graphical representation of the categorical mixture model.

3.1.3 Latent Dirichlet allocation

LDA extends CMM by grouping the observed words into M batches given by M documents. Instead of parameterizing the entire set of words with one mixture of categorical distributions, LDA parameterizes the documents with a categorical distribution with parameters θ_m $m \in 1 \cdots M$. The parameters represent the proportion of the topics assigned to each document. We depict this information in figure 3.2. As an interesting connection, CMM is equivalent to LDA for the case in which we assume the existence of only one document.

Concretely, LDA assumes that each document has a categorical distribution over K topics with a probability distribution $p(Z = k|\theta_m)$ $k \in 1K$. Each topic represents a categorical distribution over the vocabulary of words with probability $p(w|\phi_{Z_k})$. The per-document topic parameters θ_d are generated by sampling from a Dirichlet distribution with hyperparameters α , and the per-topic word parameters ϕ_w is generated by sampling from a Dirichlet distribution with hyperparameters β . The complete generative process of LDA is presented in Algorithm 3.

Using the description of LDA, it is relatively easy to estimate the probability of a word given the parameters of a document. This probability is given by:

$$P(w|\theta) = \sum_{k=1}^K P(w|\phi_k)P(Z = k|\theta) \quad (3.3)$$

Algorithm 1 LDA Generative Process

Input M documents with N_m words in document m .
Input K topics.

```

1: procedure LDASAMPLE
2:   Draw a distribution over topics  $\theta \sim \text{Dirichlet}(\alpha)$ 
3:   for each topic  $i$  in  $1 \cdots K$  do
4:     Draw a distribution over the words  $\phi_i \sim \text{Dirichlet}(\beta)$ 
5:   end for
6:   for each document  $d$  in  $1 \cdots M$  do
7:     for each word  $j$  in  $1 \cdots N_m$  do
8:       Draw simultaneously a topic  $z_{mj} \sim \text{Categorical}(\theta)$ 
9:       Draw a word  $w \sim \text{Categorical}(\phi_{z_{mj}})$ 
10:    end for
11:  end for
12: end procedure

```

Fig. 3.2 Graphical representation of the latent Dirichlet allocation model.

However, estimating the probability of a document given a word is something that LDA was not designed to model. Such probability may be useful in areas such as information retrieval. The single directionality of LDA is given by the way that the data is clustered. In a standard configuration, the document is represented by a distribution over topics and the topics are defined as distribution over words. Therefore the documents are represented as clusters of words. We depict this information in figure 3.2.

Interestingly, the relationship between documents and topics could be interchanged so that the words are modeled with a mixture distribution over topics, and the topics are defined as a distribution over the set of documents. In this case, the model would describe the probability of obtaining a document given a word, and the topics would

represent a clustering over the set of documents. These two approaches to clustering intuitively represent two different ways to summarize the same information. In Chapter 4, we present how we can mine biclusters by combining both clustering approaches into one model.

3.1.4 Biclustering

In data mining, clustering corresponds to the task of assigning the data to groups. Representing the data with clusters causes information loss but it achieves simplification and it allows discovering the structure of the data. Clustering is applied to any dataset that can be represented by a matrix where the rows correspond to the set of variables that we want to cluster and the columns correspond to the set of attributes in the dataset. Every cell in the matrix corresponds to the value of one of the attributes assigned to one of the variables. Clustering algorithms group the variables by measuring the similarities between the entire set of attributes.

In some datasets, only a subset of the attributes influences the similarity between some of the variables. Therefore, it may be convenient to study which attributes influence which variables. Attempting to cluster these datasets with the entire set of attributes may lead to diffuse results [49]. Under this motivation, biclustering algorithms have been developed to cluster simultaneously the attributes and the variables into biclusters. A bicluster represents a subset of attributes that is highly related with a subset of the variables. The most common datasets where biclustering has been applied includes mobility data, microarray data and text data. In Table 3.1, we present a comparison between clustering, biclustering and the motivation to use biclustering.

Intuitively, the extracted biclusters are short descriptions of the interdependencies between two variables [49]. Given a matrix as the input, a bicluster is represented by a

Dataset	Clustering	Motivation	Biclustering
Check-in	Groups the users based on how frequent the visit the entire set of places.	A subset of the users might commonly be found in a subset of the places. E.g., a user who is a student might visit places inside a school.	We obtain groups of users that visit a sets of places.
MicroArray	Group the genes based on their expression level across all the biological samples.	Even though all samples may be selected from the same part of the body, the cell's physiological state among other conditions might lead to some samples being unable to identify the typical expression of the cells extracted from such body part.	We obtain groups of genes that identify the expression pattern of tumor cells and we also obtain the subset of the samples that support this evidence.
Text	Group the authors based on the similarities between the words that they use to write their papers.	Authors use different words depending on the target audience of their work.	We obtain groups of authors who write about similar 'topics'. In this case, a 'topic' represents a group of words.

Table 3.1 Comparison between clustering and biclustering in three different scientific fields.

subset of the rows and a subset of columns that exhibit some relationship. Many types of relationships have been studied in the literature [81]. For example, a submatrix whose values are constant is known as the constant bicluster and a submatrix whose rows are linearly dependent is known as the perfect additive bicluster. The most common types of biclusters that have been studied are presented in Section 2.2.3.

In count data and datasets where the magnitude of a value indicates the importance of the relationship between two variables, it is indispensable to account for the magnitude of the values between variables. This is because a high magnitude represents a strong evidence of the relationship between a pair of variables. For instance, in mobility data the users tend to visit their preferred places. In text data, the authors tend to use the words of their preference. In microarray data, a gene might respond to pathogens or other organisms when its value has a high magnitude. Therefore, it is required that we extract biclusters that maximize the sum of its values. We refer to such type of biclusters as *heavy biclusters*.

For illustration, we present a mobility dataset in Table 3.2. The rows correspond to the set of users, and the columns correspond to the set of places. The values in each cell of the table correspond to the number of times that a user visited a place. The expected biclusters corresponds to the sub-matrices that include the higher values across the entire matrix. We visualize them in the table using blue and red color respectively. We observe that users U_1 , U_2 and U_3 tend to visit frequently places P_1 , P_2 and P_3 . Therefore, they are assigned to one bicluster. Similarly users U_3 , U_4 and U_5 tend to visit places P_4 , P_5 and P_6 with a high frequency and they are assigned to another bicluster. We observe how user U_3 visits all the places in the dataset and thus it is expected that he may be assigned to both biclusters. On the contrary, there is not enough visits recorded for user U_6 , and thus he should not be assigned to any bicluster. A similar case occurs for Place P_7 , however a robust model might assign

	P_1	P_2	P_3	P_4	P_5	P_6	P_7
U_1	20	22	23	0	0	1	0
U_2	17	20	18	0	0	2	0
U_3	21	19	19	10	13	7	0
U_4	0	0	0	12	9	8	15
U_5	0	1	0	11	11	7	0
U_6	0	0	2	0	3	0	0

Table 3.2 Sample mobility dataset for 6 users and 7 places. The expected biclusters are presented in blue and red.

partial membership for P_7 to the red bicluster. Furthermore, it is desirable to extract additional information about these relationships. For instance, the importance of each user and place in each bicluster and the importance of each bicluster. In the example, we observe that P_6 is not as typical as the other places in the red bicluster. Also, the blue bicluster is more important than the red bicluster because it has a stronger dependence as indicated by a higher number of visits.

In summary, it is important to discover these inter-dependent relationships and the structures of the biclusters should be flexible, i.e., it is important that the model supports cases where a row and column may be assigned to (1) only one bicluster (2) many biclusters, and (3) no bicluster. Furthermore, it is desirable to discover the importance of each row/column in biclusters, and the importance of the discovered biclusters. It is also beneficial that the algorithm does not assume the specification of the total number of biclusters a priori.

Some existing biclustering algorithms are able to extract this type of bicluster. However, these techniques typically do not support a flexible assignment of columns and rows to the biclusters and they assume the number of biclusters is known in advance. For example, Tanay et al. took a binary matrix as the input [96], Dhillon enforced that every row and column must be assigned to one bicluster [28], Wang et al., assumed that every cell in the matrix must be part of one bicluster, and Mallela et al., assumed a hard clustering [29]. More discussions can be found in Chapter 2.

We conclude that current methods impose constraints that limit the discovery of the underlying relationships in the datasets.

With the objective of relaxing the constraints imposed by related algorithms, we propose a novel algorithm for mining heavy biclusters from bivariate categorical data using a probabilistic mixture model. We construct the model by observing that a CMM may cluster the rows without taking into account the information of the columns or it may cluster the columns without taking into account the information of the rows. Nonetheless, we cannot do both at the same time despite they take the same input data. Mutual reinforcement of both approaches has shown improvement over a single approach in non-probabilistic clustering in the mobility domain [5] and the text domain [28]. In Chapter 4, we will present a novel topic model that combines the two approaches to clustering.

3.1.5 Extensions on LDA

In this section, we present two extensions of LDA that we use to evaluate our proposed inference method in Chapter 5. First, we present the relational topic model (RTM) which is a model designed to incorporate the citations between documents to the LDA model [22]. RTM adds one more step to the generative process of LDA as follows:

Draw if there exists a link between documents d and d' $y_{d,d'} \sim \psi(\cdot | \bar{z}_d, \bar{z}_{d'})$

$y_{d,d'}$ is an indicator variable and its probability is given by the following function:

$$\psi(y_{d,d'}=1) = \exp(\eta^T(\bar{z}_d \odot \bar{z}_{d'} + \nu))$$

where η and ν are hyperparameters, $\bar{z}_d = \frac{1}{N_d} \sum_{i=1}^{N_d} z_{d,i}$ ($z_{d,i}$: one-hot based topic assignment), and $\odot$ means element-wise product.

The second model that we present is the supervised LDA (sLDA) [73]. sLDA uses document labels to fit the extraction of the topics. Concretely, sLDA extends LDA

by adding an observed variable x that selects the label of a document. For example, the rating given to a movie, or the number of likes given to a post. sLDA adds the following step to the generative process of LDA:

Draw a label x_d to be applied to document d $x_d \sim N(\gamma^T \bar{z}_d, \sigma^2)$

where both γ and σ are hyperparameters.

So far, we have presented the description two models that extend the standard LDA. However, learning the models requires an often complex process that estimates the parameters of the model. The inference of topic models is typically performed using sampling approaches or optimization approaches. In our study, we find new contributions to the Gibbs-based inference approach. Hence, we describe in the next section how to use collapsed Gibbs sampling to do inference of Topic models.

3.2 Gibbs Sampling Inference

The objective of the inference methods is to find an estimation of the parameters of the model given a dataset X . In the case of LDA, we want to estimate, θ , ϕ , and Z . Ideally, we should be able to analytically find $f(\theta|X)$, $f(\phi|X)$, and $f(Z|X)$. However, in the case of LDA, just like in many other probabilistic models, finding these probabilities requires solving intractable integrals. Therefore, we turn to methods such as Gibbs sampling to estimate the parameters. Gibbs sampling obtains samples from tractable conditional distributions which in turn can be used to estimate the desired parameters. To understand this process, we start with a simpler example.

Generally speaking, the objective of Gibbs sampling is to obtain samples for a probabilistic model with density $f(X_1, \dots, X_n)$ where $X_1, \dots, X_n$ are random variables.

Gibbs sampling guarantees that we can obtain samples $x_1 \dots x_n$ from the joint density if we iteratively perform updates using only the conditional densities:

1. Draw random initial samples x_{-1}
2. Sample x_1 from $f(X_1|X_{-1} = x_{-1})$
3. $\vdots$
4. Sample x_n from $f(X_n|X_{-n} = x_{-n})$
5. Reiterate for several iterations.

Where the subscript $_{-i}$ denotes all variables except the i^{th} variable. This method guarantees that after a number of iterations the obtained samples $x_1, \dots, x_n$ will correspond to draws from the desired joint density $f(X_1, \dots, X_n)$. One drawback of this method is that we need to know an analytic form for the conditional densities which may not apply for some probabilistic methods.

3.2.1 Convergence of Gibbs sampling

The convergence of Gibbs sampling is fully described by the theory of Markov chains. Hence, we begin our explanation by introducing the Markov chain.

A Markov chain is a stochastic model that describes the probability of observing future events given the probability of observing the current events. The model uses a transition probability matrix to compute the probability of future events. Concretely, given the probability of two events x_1, x_2 , and a transition matrix $A_{xx'}$, the probability of observing the same events one step in the future is given by $x' = A_{xx'} * x$, where x is a vector composed of the probability of the events x_1, x_2 and x' is a vector composed of the probability of the future events x'_1, x'_2 . A particular distribution used to study steady states in a Markov chain model is known as the stationary distribution. The stationary distribution refers to a distribution $\bar{x}$ such that $\bar{x} = A_{xx'} * \bar{x}$. That is, the stationary distribution is a distribution immutable after applying the transition matrix.

Interestingly, if the transition matrix does not have zeros in it, it is always possible to find the stationary distribution by initializing x_0 to be any distribution at the first iteration and then computing $A^n x_0$ to find the probability of events at step n . As n increases, the distribution of x will tend to converge to the stationary distribution.

One iteration of Gibbs sampling can be framed as one step on a system of Markov chains. One Markov chain for each random variable in the system. The explanation can be easily presented if we assume that we are running Gibbs sampling for two variables X and Y . In which case one iteration of Gibbs sampling corresponds to:

$$x_2 \sim f(X|Y = y_1) \quad (3.4)$$

$$y_2 \sim f(Y|X = x_2) \quad (3.5)$$

where y_1 is an initial chosen value. In this case, the transition from y_1 to y_2 follows the transition distribution given by:

$$f(Y^2|Y^1 = y_1) = \int_X f(Y^2|X)f(X|Y^1 = y_1) dX \quad (3.6)$$

Hence, the distribution used to draw y_2 corresponds to:

$$f(Y^2) = f(Y^2|Y^1 = y_1)f(Y^1 = y_1) \quad (3.7)$$

Given the theory of stationary distributions, we know that as the number of iterations increases, the densities for $f(X^n)$ and $f(Y^n)$ will converge to a marginal densities $f(\bar{X})$ and $f(\bar{Y})$ after a sufficiently large value for n . At this point, the iterations of Gibbs sampling will not have an improvement on the estimation of the parameters. The proof on the convergence of Gibbs sampling for a larger number

of variables does not trivially follow the case for two variables. Although, the case of two variables is enough to illustrate how to link Gibbs sampling to the theory of Markov chains, It is suggested to read the paper from George and Casella [19] for a more in-depth explanation on how to generalize this proof to an additional number of variables.

At convergence, the samples obtained from the stationary distributions correspond to samples from $f(X_1, X_2)$. As such, the samples can be used to obtain any conditional or marginal distributions from the probabilistic model.

It is important to note that the theory of Markov chains uses distributions to proof the convergence of Gibbs sampling, however during the iterations of Gibbs sampling, the densities of $f(X)$ and $f(Y)$ are unknown as the algorithm only stores samples and it does not compute densities. As a result, it is unknown if at a given iteration, the obtained samples are drawn from stationary densities. As we show in Section 3.2.3 for LDA, the convergence of Gibbs sampling is guaranteed but it is hard to assess.

3.2.2 Collapsed Gibbs sampling

The collapsed Gibbs sampler (CGS) is a special case of Gibbs sampling in which some of the conditional densities may not be necessary to sample because they are collapsed using analytical integration. As an example, we show how we can apply collapsing on LDA.

In Gibbs sampling, LDA requires that we draw samples from $f(\theta|\phi, \mathbf{Z}, X)$, $f(\phi|\theta, \mathbf{Z}, X)$, and $f(\mathbf{Z}|\theta, \phi, X)$, where X denotes the dataset. However, we may avoid computing $f(\theta|\phi, \mathbf{Z}, X)$ and $f(\phi|\theta, \mathbf{Z}, X)$ because our model $f(\mathbf{Z}, \theta, \phi)$ can be reduced by integrating θ and ϕ so that $\int_{\theta} \int_{\phi} f(\mathbf{Z}, \theta, \phi, X) = f(\mathbf{Z}, X)$. To estimate $\mathbf{Z}$, we can simply iterate over the the conditional densities $f(Z_i|Z_{-i}, X) \quad \forall i \in M * N_m$.

CGS iteratively samples the posterior $p(Z_j|Z_{-j}, \mathbb{D})$ for each record in the dataset. Note that “-j” in a quantity means that the contribution of the j^{th} record is not considered in the quantity. Then, the drawn sample $k_j(k_j : 1 \sim K)$ is used to update the statistics of the model. The statistics correspond to the sufficient functions over the sampled topics necessary to estimate the parameters of the probabilistic topic model [45]. Table 3.3 lists the posterior density of each graphical model as a function of the statistics $\Omega = \{C, D, N\}$. The statistics for some topic models are defined as follow:

- C_{k_j, d_j} : the vector of topic assignments for document w_j without considering the j^{th} record.

$$C_{k_j, d_j} = \sum_{n=1, n \neq j}^{|\mathbb{D}|} \mathbb{1}(d_j = d_n \wedge k_j = k_n) \quad (3.8)$$

- D_{k_j, w_j} : the vector of topic assignments for word w_j without considering the j^{th} record.

$$D_{k_j, w_j} = \sum_{n=1, n \neq j}^{|\mathbb{D}|} \mathbb{1}(w_j = w_n \wedge k_j = k_n) \quad (3.9)$$

- N_{k_j} : the vector of topic assignments over all the dataset without considering the j^{th} record.

$$N_{k_j} = \sum_{n=1, n \neq j}^{|\mathbb{D}|} \mathbb{1}(k_j = k_n) \quad (3.10)$$

We use the z_j to denote the one-hot vector representation of the topic assignment k_j , $\mathbb{1}()$ represents the indicator function, and $\wedge$ is the logical AND operator. This notation is convenient to contrast the differences CGS and additional inferences methods presented in this document.

Table 3.3 Posterior density as a function of the statistics Ω

Model	Unnormalized posterior density $g(\Omega)$
LDA	$(C_{k_j, d_j}^{-j} + \alpha) \frac{D_{k_j, w_j}^{-j} + \beta}{D_{k_j, \cdot}^{-j} + D \beta}$
RTM	$(C_{k_j, d_j}^{-j} + \alpha) \frac{D_{k_j, w_j}^{-j} + \beta}{D_{k_j, \cdot}^{-j} + D \beta} \exp\left(\frac{b_k}{N_d} \sum \frac{n_{d', k}}{N_{d'}}\right)$
sLDA	$(C_{k_j, d_j}^{-j} + \alpha) \frac{D_{k_j, w_j}^{-j} + \beta}{D_{k_j, \cdot}^{-j} + D \beta} \exp\left(-(y_d - b^T \bar{z}_d - a)^2\right)$

Algorithm 2 describes the unified sampling method by sequentially processing each of the $|\mathbb{D}|$ records in the dataset.

The unified CGS algorithm shares some commonality across several graphical models. It first computes the statistics for each model, which are required to compute the posterior equation $g(\Omega)$ for each record in the dataset.

Therefore, as shown by line 3, the algorithm initially computes the statistics $\Omega = \{C, D\}$ for each model. Rather than recomputing all the parameters of the statistics for the update of the topic assignment for each record in the dataset, it is more efficient to sufficiently update the statistics so that the definitions given by Equations 3.8, 3.9, and 3.10 still hold. In line 6, CGS subtracts the contribution of the j^{th} record from each of the statistics before computing the conditional distribution $g(\Omega)$. Then, at line 9, it updates the statistics with the new topic assignment z'_j .

For completeness, Equation 3.11 shows the update of the statistics of the LDA model at the j^{th} record:

$$C_{k_j, d_j} = C_{k_j, d_j} + 1 \quad ; \quad D_{k_j, w_j} = D_{k_j, w_j} + 1 \quad (3.11)$$

The derivation for the equations in Table 3.3 are presented in Jonathan Chang's thesis [21]. Also, a detailed derivation of LDA is given by Bob Carpenter [18]

Algorithm 2 Collapsed Gibbs Sampling (CGS)

```

1: procedure CGS
2:   Input: Dataset  $\mathbb{D}$  (size:  $|\mathbb{D}|$ ), latent assignment  $z_j$ 
3:   Initialize  $z_j, \forall z_j \in 1 \dots |\mathbb{D}|$ ,  $\Omega = \text{computeStats}(z_j)$   $\Omega = \{C, D\}$ 
4:   repeat
5:     for  $j = 1$  to  $|\mathbb{D}|$  do
6:        $\text{updateStats}(\Omega, z_j, \text{Subtract})$ 
7:        $p(Z_j | Z_{-j} = z_{-j}, \mathbb{D}) \propto g(\Omega)$  from Table 3.3
8:        $z'_j = \text{sample}(p(Z_j | Z_{-j} = z_{-j}, \mathbb{D}))$ 
9:        $\text{updateStats}(\Omega, z'_j, \text{Add})$  from Equation 3.11
10:    end for
11:  until convergence
12: end procedure

```

At convergence, we can use the topic assignments to estimate ϕ and θ . We present the estimation of θ and ϕ in the next section.

3.2.3 Parameter estimation of the latent Dirichlet allocation

The latent topic assignments are all the information necessary to find the parameters ϕ and θ . Typically, the algorithm runs for a large number of iterations. Then, the topic assignments are stored to estimate the parameters θ and ϕ . The samples are stored at spaced intervals to simulate independent sampling [95]. Given S samples $z_j^s \forall j \in 1 \dots M \forall s \in 1 \dots S$, for a total of $S \times M$ samples. We can estimate the parameters as:

$$\hat{\theta}_{x_j^{(d)}} \propto \frac{1}{S} \sum_{s=1}^S (D_{x_j^{(d)}}^s + \alpha) \quad ; \quad \hat{\phi}_{x_j^{(w)}} \propto \frac{1}{S} \sum_{s=1}^S (W_{x_j^{(w)}}^s + \alpha) \quad (3.12)$$

Where $D_{x_j^{(d)}}^s$ and $W_{x_j^{(w)}}^s$ are constructed from the samples obtained by the s^{th} sampling batch.

Although the convergence of LDA-CGS is guaranteed [19], the assessment of convergence must be estimated and the accuracy of this estimation depends on the

method used to obtain the samples as well as the number of samples taken for the estimations of $\hat{\theta}$ and $\hat{\phi}$ [114].

As show in Section 3.2.1, the problem stems from sampling. The samples represent all the information about the marginal densities and it is hard to measure if the Markov chains are approximating the target marginal distributions $p(Z_j) \forall j \in 1 \cdots M$. In our studies, we develop an approach more closely connected to the theory of Markov chains where instead of sampling, we use the entire distribution when doing one step on the Gibbs inference method. We present this approach in Section 5.1.

Additional problems of the collapsed Gibbs sampling inference are revealed explicitly by Algorithm 2. First, the statistics used to compute the posterior in line 6 are updated after every tuple in the dataset. As a consequence, the algorithm is non-distributable. Although approximate methods have been proposed to solve the distributability of the inference, they are only applicable to some specific graphical models and they sacrifice the predictive power of the collapsed Gibbs sampler [93, 108]. In addition, lines 6 to 9 show that the posterior must be computed and sampled for every row in the dataset, even when multiple tuples are repeated. The repetition of tuples happens every time a word appears multiple times in a document. Text datasets have around 20% repeated tuples [89]. In Section 5.2, we present a novel deterministic distributable inference method which is able to avoid the computation of repeated tuples.

3.3 Summary

In this chapter, we presented with high level of detail the relevant work and theory necessary to understand the main contribution of the thesis which will be presented in the following chapters of this thesis. In Chapter 4, we present a novel probabilistic topic model for the mining of biclusters or simultaneous clustering of two random variables. For instance, the authors of documents and the words written inside their documents.

Rather than grouping the words into topics, or grouping the authors by their use of words, biclusters enable the simultaneous clustering of the relationship between the authors and their use of words. Such biclustering enables the solution to the question “Which groups of authors write about which topics?”. This information is useful for applications such as reviewer recommendation. For a full understanding of Chapter 4, it is imperative to understand well the theory behind topic models. Therefore, we have presented on this chapter the fundamental theories about probabilistic Bayesian models. In particular, we presented a formal definition of topic models starting from the Bayesian principle of modeling the uncertainty using prior distributions. Then, we showed how this principle is used to construct the categorical mixture model, which is arguably the most basic model for clustering a discrete random variable and finally, we showed how topic models use the categorical mixture model as the building block to extract the topics of a corpus.

The understanding of how to use random variables to model datasets and how to extract information is essential for designing an useful probabilistic model. However, learning how to find the parameters of a proposed model is perhaps a more important task. This is because, it is arguably easier to propose probabilistic distributions to model a dataset as compared to solving the mathematics required to fit the parameters of a model to a given dataset. Fitting the parameters of a probabilistic model is known as the inference process. In theory, the inference of a probabilistic model requires finding integrals that in a large number of cases cannot be solved numerically. Therefore, researchers have proposed several approximation algorithms that although they work well in practice, they have undesirable properties such as high computational complexity and divergence from the expected result. As a consequence, the study about the inference of topic models is an active branch of research with one of the main subbranches being sampling algorithms.

On Chapter 5, we present a contribution to the field of the inference of topic models. Our contribution is based on Gibbs sampler and therefore, to understand our proposed method, we presented on this Chapter the theory of Gibbs sampling. In particular, we showed how a Gibbs sampler uses a Markov chain as its building block. We also included in our explanation why Gibbs sampling converges towards the expected result and why its convergence is guaranteed. Then, we showed how some models allow simplifying Gibbs sampling by collapsing some of the random variables and finally, we presented how the collapsed version of Gibbs sampling can be applied to the latent Dirichlet allocation model.

Chapter 4

Dual Topic Models for Biclustering

In this chapter, we present and solve the first research problem which corresponds to the explanation of how we can use dual topic models to mine biclusters. We start by introducing the advantages that topic models bring to the biclustering extraction problem. Then, we explain how we can use the Categorical Mixture Model (CMM) to construct a model with these advantages. We labeled our method as the dual topics to bicluster model (DT2B). We formally present the generative process and the posterior equation that describes how we find the parameters of our proposed model using collapsed Gibbs sampling. Then, we use the posterior distribution to prove that DT2B corresponds to a generalization of LDA. Finally, we provide an explanation for why our proposed model extracts biclusters followed by an experimental evaluation to showcase the advantages of our model in Section 4.7.4.

4.1 Problem Definition

Madeira et al., gave the following definition of biclustering [70]. *The goal of biclustering methods is to identify subgroups of rows and subgroups of columns in a matrix, by*

performing simultaneous clustering of both dimensions, instead of clustering these two dimensions separately.

In biclustering, we can inherit the following advantages from mixture models:

Advantage 1. Since biclustering clusters the rows and the columns of a matrix, we may use a latent variable to cluster the rows into row-topics and another latent variable to cluster the columns into column-topics. From this approach to modeling, we can obtain information about which rows are relevant in each row-topic and which columns are relevant in each column-topic.

Advantage 2. If we mix every latent variable independently, we would obtain two mixture models clustering independently each of the dimensions. However, our goal is to extract biclusters that represent a strong relationship between the rows and the columns. To achieve this, we model the dependency of the two latent variables. In other words, the mixture will encode which row-topics are related to which column-topics.

Advantage 3. Being the model a Generative model, we obtain description of how the data is generated. Therefore, we can use the model to solve additional problems such the extraction of the joint and the conditional distributions between the rows and the columns.

Advantage 4. Due to the soft-clustering nature of probabilistic approaches, our model does not limit to a particular biclustering structure.

For the ease of presentation, we describe this chapter using the relationship between users and the places they visit, rather than the relationship of words written in documents. Formally, given a matrix $|U| \times |P|$, we may use a categorical mixture model to cluster the set of rows U by using the information in all the columns. The mixture model would represent the rows as random mixture of latent topics Z_u , where each topic is characterized by a distribution over the rows, denoted by ϕ_u .

Based on the same matrix, we may use a second categorical mixture model to cluster the set of columns P by using the information in all the rows. This mixture model would represent the columns as random mixture of latent topics Z_p , where each topic is characterized by a distribution over the columns, denoted by ϕ_p . We refer to this mixture model as the Dual Mixture model since we use the same input data to obtain a complementary result. We label the topics obtained by the mixture model as *row-topics* and the topics obtained by the dual mixture model as *column-topics*.

We need to combine the mixture model with its dual in order to perform simultaneous clustering of both dimensions. Hence, we must model a mutual dependency between the row-topics Z_u and the column-topics Z_p . We may use additional parameters $\theta_{i,j}$ to capture the relationship between the i -th row-topic and the j -th column-topic.

Using the sample dataset presented in table 3.2, we aim to find the groups of users as user-topics, the groups of places as place-topics and the relationship between each user topic and each place topic. Those relationships with higher probability will correspond to the extracted biclusters.

The groups of users will be obtained from Z_u . So $Z_u = 1$ will be characterized by the following user-topic:

$$\phi_{u(1)} = [0.36, 0.32, 0.34, 0, 0, 0.01]$$

Similarly $Z_u = 2$ will be characterized by:

$$\phi_{u(2)} = [0, 0, 0.34, 0.33, 0.33, 0]$$

The groups of places will be obtained from Z_p , So $Z_p = 1$ and $Z_p = 2$ will be characterized by the following place-topics:

$$\phi_{p(1)} = [0.33, 0.33, 0.34, 0, 0, 0, 0]$$

$$\phi_{p(2)} = [0, 0, 0, 0.32, 0.32, 0.26, 0.1]$$

Because θ must encode the relationship between every user-cluster and every place-cluster, we expect to find the following values:

$$\theta = \begin{bmatrix} 0.60 & 0.05 \\ 0.05 & 0.30 \end{bmatrix}$$

If we consider a relationship to be a bicluster when $P(Z_u, Z_p | \theta) > 0.25$, then we would extract two biclusters. The first bicluster corresponds to the group of users extracted from the first user-topic $Z_u = 1$ and the group of places extracted from the first place-topic $Z_p = 1$, while the second bicluster corresponds to the groups of users extracted from the second user-topic $Z_u = 2$ and the group of places extracted by the second place topic $Z_p = 2$. We may use a similar criteria to determine which users are extracted from the user-topic and which places are extracted from the place-topic. For instance, in the first bicluster we may select a user to be part of the bicluster if $P(u | \phi_{u(1)}) > 0.25$. In this case U_1 , U_2 and U_3 will be part of the bicluster. For the places, we may select a place to be part of the bicluster if $P(p | \phi_{p(1)}) > 0.25$. In this case P_1 , P_2 and P_3 are part of the bicluster.

In conclusion, we aim to use row-topics Z_u characterized by the parameters ϕ_u to obtain groups of rows and the information about which rows are more important in each bicluster. Similarly, we aim to use column-topics Z_p characterized by the parameters ϕ_p to obtain groups of columns and the information about which columns are more important in each bicluster. Finally, we expect to have some parameters θ from where

we can obtain the information about the relationships between the row-topics and the column-topics. Successively, we may extract those relationships with higher probability as biclusters.

4.2 LDA and the Dual LDA

In this subsection, we present how we designed Dual Topics to Bicluster model based on exploiting the mono-directional modeling approach of the LDA and its dual. Then, we present how we construct our model using the Categorical Mixture Model (CMM), similarly to how CMM is used to construct LDA model. Then, in the next section, we will explain how we use two CMM models to extract both user-topics and place-topics in a single model.

Topic models allow us to have a single directionality of modeling a categorical dataset. This consequence can be easily explained using mobility data where the users take on the role of the documents, and places take on the role of the words. Under this setting, we explicitly model the probability $p(p|u)$. Hence, LDA's topics represent distributions over the set of places or *Place-topics*.

However, LDA hinders the representation of the distribution of users visiting any of the places $p(u|p)$ which gathers explicit information about grouping the set of users. This limitation of an LDA can be solved using a *dual LDA*. The dual LDA is also an LDA model applied to the same data, but it interchanges the role of the users and places. In the dual LDA, the distribution of users visiting any of the places $P(u|\theta_p)$ is directly represented by the dual model and the topics represent distributions over the set of users. A graphical representation of the information extracted by an LDA and its Dual LDA is presented in figure 4.1.

Fig. 4.1 Information extracted by an LDA (left) and a Dual LDA (right). The width of the arrow represents the preference of a topic. The ovals represents the topics.

In summary, we can express the relationship between users and places into two approaches.

i) LDA or individual based-approach:

- A user is defined by the places that he visited.
- We obtain topics represented by distributions over the set of places.
- We can represent the distribution of the places for a given user.

ii) Dual LDA or location based-approach:

- A place is defined by the users who visit the place.
- We obtain topics represented by distributions over the set of users.
- We can represent the distribution of the users for a given place.

4.3 Dual Topics to Bicluster (DT2B)

4.3.1 Intuitions

We propose a new way of modeling the relationship between the two random variables. Instead of describing the distribution of places given a user $p(p|u)$, or the distribution

Fig. 4.2 Information extracted by DT2B. The ovals represents the topics and the width of the arrows represents the strength of the relationship between two topics.

of users given a place $p(p|u)$, we propose a description of the joint distribution of the two variables $p(p, u)$. If we obtain a description of the joint distribution, we can quickly derive the conditional distributions modeled by the LDA and the dual LDA. In our model, we want to extract place-topics as done by the LDA and we also want to extract user-topics as done by the dual LDA. As a novelty, we also extract the mutual dependency between the topics. This dependency will provide the information about which user-topics are related to which place-topics (Figure 4.2). In conclusion, we obtain the following intuitions:

1. Group the users into user-topics.
2. Group the places into place-topics.
3. Extract the mutual dependency between the place-topics and the user-topics.

4.3.2 Using CMM to implement intuitions 1 and 2

We construct the dual topics to bicluster model by implementing the three intuitions. For the implementation of intuitions 1 and 2, we use the Categorical Mixture model presented in subsection 3.1.2 since we know that the CMM can extract topics of a discrete variable. Therefore, we use one CMM to extract user-topics and another CMM

t

Fig. 4.3 Implementation of intuitions 1 and 2.

for extracting place-topics. A graphical representation of the model implementing only intuitions 1 and 2 is presented in figure 4.3.

4.3.3 Defining the bivariate categorical distribution to implement intuition 3

Describing the dependency between two kinds of topics presents a challenge because it has not been studied by Topic Models. We address this challenge by defining a probability distribution to measure the probability of observing a user-topic and a place-topic. Because we require a probability distribution to describe the simultaneous generation of two variables, we define a bivariate probability distribution. Additionally, it would be ideal to use a categorical distribution because categorical distributions are typically used to capture the mixing components of the topics. With a categorical distribution, we can use the parameter $\theta_{i,j}$ to capture the relationship from the i -th user-topic to the j -th place-topic. Given these conditions, we define the *bivariate categorical distribution*.

In order to define a probability distribution, we need to specify the support, parameters and probability mass function of the distribution. The support of a

probability distribution represents the set of possible values of a random variable having that distribution, the parameters corresponds to the set of variables that describe the probability mass function and the probability mass function represents how we assign a probability to every element in the support of the distribution. Using this components, we present the following description of the bivariate categorical distribution:

Support The distribution has a two dimensional support since we require simultaneous sampling of two random variables.

Parameters The distribution uses $K_u \cdot K_p$ parameters. Each parameter represents the relationship between one place-topic and one user-topic.

Probability Mass Function $f_\theta(\vec{x}) = \sum_i^{K_u} \sum_j^{K_p} \mathbb{1}[x_0 = i \wedge x_1 = j] \theta_{i,j}$ This function represents the probability of observing every user-topic, place-topic pair $(\vec{x})$ using a parameter.

Now we present the formal definition of the bivariate categorical distribution:

Definition 1. A bivariate categorical distribution $Categorical^2(\theta, K_u, K_p)$ is a discrete probability distribution parameterized by $\theta \in \mathbb{R}_{[0,1]}^{K_u \cdot K_p}$ and $K_u, K_p \in \mathbb{N}_{>0}$. The support of the distribution is $X \in \mathbb{N}_{[1:K_u]} \times \mathbb{N}_{[1:K_p]}$. The probability mass function is given by $f_\theta(\vec{x}) = \sum_i^{K_u} \sum_j^{K_p} \mathbb{1}[x_0 = i \wedge x_1 = j] \theta_{i,j}$

Proof. We proof that the bivariate categorical distribution is a valid distribution by showing how we can represent the bivariate categorical distribution in terms of the categorical distribution:

Given a random variable $x \sim Categorical^2(\theta, K_u, K_p)$. We need to use a mapping g so that $y = g(x) \sim Categorical(\theta)$. This mapping is given by $g : \mathbb{N}_{[1:k]} \rightarrow \times \mathbb{N}_{[1:K_u]} \times$

Fig. 4.4 Function used to define the *Categorical*² distribution. The visualization indicates the mapping from the parameters to the support set.

$\mathbb{N}_{[1:K_p]}$ where $g(x) = (x \div K_p, x \bmod K_p)$. We present in Figure 4.4 a visualization of the mapping of g . Because g is a bijection, we are guaranteed that we can always represent a bivariate categorical distribution in terms of the categorical distribution and vice versa.

■

4.3.4 Model representation

By connecting the implementation of the above-mentioned intuitions, we obtain the dual-topics to bicluster (DT2B) model. DT2B assumes a known dimensionality K_u of user-topics and K_p of place-topics. In figure 4.5, we present the graphical model that encodes the dependencies between the random variables used in DT2B. We also present the formal definition of all the variables used by our model in table 4.5.

From the structure of the model, we observe that we can obtain the joint probability of observing the users and places as given by:

$$P(u, p) = \sum_{Z_u=1}^{K_u} \sum_{Z_p=1}^{K_p} P(u|\phi_{Z_u})P(p|\phi_{Z_p})P(Z_u, Z_p|\theta) \quad (4.1)$$

Fig. 4.5 Dual topics to bicluster graphical model

4.4 Generative and Inference Process

The generative process uses the intuitive idea of LDA that users are drawn from user-topics. From the Dual LDA, it uses the intuitive idea that places are drawn from place-topics. To connect both models, DT2B uses an extra latent variable (θ) from which a place-topic and a user-topic are sampled simultaneously. This latent variable contains the information of the relationship between every user-topic and every place-topic. In algorithm 3, we present the algorithmic description of the generative process.

Another challenge of the DT2B is the inference of the posterior distribution $P(z_u, z_p | u, p)$. This distribution is intractable to compute due to the fact that the topics are dependent on each other. Furthermore, applying commonly used inference techniques such as variational inference or collapsed Gibbs sampling is not viable since the generative process that we describe requires a simultaneous sampling. This type of sampling has not been studied in topic modeling.

Equation 4.2 corresponds to the posterior distribution that we use to update the user-topic $z_{u(v)}$ and place-topic $z_{p(v)}$ assigned to the v -th visit record. For each visit

Variable	Dimension	Description	Distribution
v	1	Index to a visit record	NA
V	1	Number of places	NA
K_p	1	Number of place-topics	NA
K_u	1	Number of user-topics	NA
U	1	Number of Users	NA
P	1	Number of Places	NA
ϕ_p	$P \times K_p$	Place per place-topic distribution	$Dirichlet(\beta_p)$
β_p	$P \times 1$	Hyperparameter of ϕ_p	NA
p_v	1	Observed place in the v -th visit record	$Categorical(\phi_p, K_p)$
ϕ_u	$U \times K_u$	User per user-topic distribution	$Dirichlet(\beta_u)$
β_u	$U \times 1$	Hyperparameter of ϕ_u	NA
u_v	1	Observed user in the v -th visit record	$Categorical(\phi_u, K_u)$
θ	$K_u \times K_p$	Joint distribution of user-topic and place-topics	$Dirichlet(\alpha)$
α	$K_u \times K_p$	Hyperparameter of θ	NA
$z_{u(v)}, z_{p(v)}$	1,1	User-topic and Place-topic assigned to the v -th visit record	$Categorical^2(\theta, K_p, K_u)$

Table 4.1 Variable definitions for the dual topics to bicluster model.

Algorithm 3 DT2B Generative Process

```

1: procedure DT2BSAMPLE
2:   Draw a distribution over topics  $\theta \sim Dirichlet(\alpha)$ 
3:   for  $i$  in  $1 \dots K_u$  do
4:     Draw a distribution over the users  $\phi_i \sim Dirichlet(\beta_u)$ 
5:   end for
6:   for  $j$  in  $1..K_p$  do
7:     Draw a distribution over the places  $\phi_j \sim Dirichlet(\beta_p)$ 
8:   end for
9:   for  $v$  in  $1..|V|$  do
10:    Draw simultaneously a user-topic and a place-topic  $z_{p(v)}, z_{u(v)} \sim$ 
       $Categorical^2(\theta)$ 
11:    Draw a user  $u \sim Categorical(\phi_{z_{u(v)}})$ 
12:    Draw a place  $p \sim Categorical(\phi_{z_{p(v)}})$ 
13:   end for
14: end procedure

```

record, we must sample from this distribution to update its topic assignments, the posterior distribution must be recomputed after each assignment. Updating all the records corresponds to one iteration. After several iterations, the posterior distribution will converge, and we have found the topic assignment for each record. The derivation of the posterior equation is presented in Section A.

$$P(z_{u(v)}, z_{p(v)} | z_{u(-v)}, z_{p(-v)}, u, p, \alpha, \beta_u, \beta_p) \propto \frac{\mathcal{Z} \mathcal{P}(v) \mathcal{U}(v)}{\sum_{j=1}^{|P|} \mathcal{P}(j) \sum_{i=1}^{|U|} \mathcal{U}(i)} \quad (4.2)$$

$$\begin{aligned} \mathcal{Z} &= \sum_{m=1}^{|U|} \sum_{n=1}^{|P|} c(z_{p(v)}, z_{u(v)}, m, n)^{(-v)} + \alpha_{z_{u(v)}, z_{p(v)}} \\ \mathcal{P}(i) &= \sum_{y=1}^{Kp} \sum_{n=1}^{|P|} c(z_{p(v)}, y, p(i), n)^{(-v)} + \beta_{p(i)} \\ \mathcal{U}(i) &= \sum_{x=1}^{Ku} \sum_{m=1}^{|U|} c(x, z_{u(v)}, m, u(i))^{(-v)} + \beta_{u(i)} \\ c(x, y, m, n) &= \sum_{j=1}^{|V|} I(z_{u(j)} = x, z_{p(j)} = y, u(j) = m, p(j) = n) \end{aligned}$$

The inference equation depends on three types of counting. First, a normalized counting of the user assignment to the user-topics $\mathcal{U}$ capturing user co-occurrence. Second, a normalized counting of the place assignment to the place-topics $\mathcal{P}$ capturing place co-occurrence. And finally a counting over the current relationship between user-topics and place-topics $\mathcal{Z}$.

4.5 Connection with LDA and its Dual

We start by presenting the inference equation used to update the topics in the LDA model. Then, we compare with the inference equation obtained by the DT2B model. In conclusion, we find that DT2B generalizes LDA and its dual.

4.5.1 Inference of LDA

The topic assignment to the v – th visit record $\{u(v), p(v)\}$ is given by sampling from the following derived posterior probability [13]:

$$P(z_{p(v)} | z_{p(-v)}, u, p, \alpha_u, \beta_p) \propto \frac{\bar{\mathcal{Z}} \mathcal{P}(v)}{\sum_{j=1}^{|P|} \mathcal{P}(j)}$$

where

$$\bar{\mathcal{Z}} = \sum_{n=1}^{|P|} c(z_{p(v)}, u(v), n)^{(-v)} + \alpha_{z_{p(v)}}$$

$$\mathcal{P}(i) = \sum_{m=1}^{|U|} c(z_{p(v)}, m, p(i))^{(-v)} + \beta_{p(i)}$$

$$c(x, m, n) = \sum_{j=1}^{|V|} \mathbb{1}(z_{u(j)} = x, u(j) = m, p(j) = n)$$

Since the function c corresponds to sums of indicator functions, it is easy to interpret the two factors responsible for the topic assignment:

- i) $\bar{\mathcal{Z}}$: Counting of topics assigned to user u_v .
- ii) $\frac{\mathcal{P}(v)}{\sum_{j=1}^{|P|} \mathcal{P}(j)}$: Probability of observing the place p_v given a topic.

4.5.2 Comparison with the inference equation of DT2B

We show that LDA is a particularization of DT2B by proving that LDA's inference equation can be reduced to the inference equation of DT2B when we apply the condition that each user is assigned to a single and unique user-topic. i.e., users and user-topics are assigned in a *one-to-one* mapping

Proof. Because we are given the user-topic assignments for every user, the inference of DT2B only depends on $z_p(v)$. So, recalling equation 4.2, we begin by analyzing the term $\frac{\mathcal{U}(v)}{\sum_{i=1}^{|U|} \mathcal{U}(i)}$. Because $\mathcal{U}(v)$ does not depend on $z_p(v)$, we can replace $\mathcal{U}(v)$

with a constant value. Then, we analyze the term $\mathcal{Z}$. Since $z_u = u \ \forall u \in 1 : U$, we can replace z_u with u and as result, we end up with the LDA inference equation. That is:

$$P(z_{p(v)} | z_u, z_{p(-v)}, u, p, \alpha, \beta_u, \beta_p) =$$

$$P(z_{p(v)} | z_{p(-v)}, u, p, \alpha_u, \beta_p) \propto \frac{\bar{\mathcal{Z}} \mathcal{P}(v)}{\sum_{j=1}^{|P|} \mathcal{P}(j)} \text{ where}$$

$$\bar{\mathcal{Z}} = \sum_{m=1}^{|U|} \sum_{n=1}^{|P|} c(z_{p(v)}, u(v), m, n)^{(-v)} + \alpha_{u(v), z_{p(v)}}$$

and α_u corresponds to a K_p dimensional vector conditioned on the user u . In LDA, the hyperparameter of the Dirichlet prior on the per-user topic distributions $\bar{\alpha}$ is independent of the given user to avoid over-fitting. We can break the dependence of α and the given user by setting α to the same value for all users. ■

Finally, we note that if we start off by assigning a *one-to-one* mapping between the places and the place-topics, we would end up with the inference updating equation of the dual LDA.

4.6 Bicluster Extraction using DT2B

If we do not implement intuition three presented in Section 4.3.3, DT2B would obtain a clustering of the set of users as given by the user-topics. Independently, DT2B would obtain a clustering model of the set of places as given by the place-topics. By enforcing a mutual dependency between the user-topics and the place-topics, we obtain an algorithm capable of identifying relationships between subsets of the set of places and subsets of the set of users. The relationship between a group of users and a group of places corresponds to a bicluster.

In Section 2.2, we have introduced the problem of biclustering as well as the categorization of the biclustering algorithms by types and structures. Using DT2B, we don't fit the structure of the biclusters and so we can extract the structures presented in figures 2.2a, 2.2b, 2.2c, 2.2d. With respect to the biclustering types, DT2B can extract those biclusters that approximate to the bicliques of the dataset. We will give a formal explanation on how does the model extract bicliques in Section 4.7.

We obtain the biclusters from the parameters of the DT2B model. First, we can obtain the groups of users from the parameter ϕ_u . Second, we can obtain the groups of places from the parameter ϕ_p . Finally, we can obtain the biclusters from the parameter θ , because θ represents the relationship between every place-topic and every user-topic.

Because the parameters ϕ_u represent $|K_u|$ distributions over the whole set of users and ϕ_p represents $|K_p|$ distributions over the whole set of places. Then, we need to establish thresholds in order to decide which user's are more relevant in a user-topic, which places are more relevant in a place-topic and which relationships are the strongest relationships between user-topics and place-topics.

We extract the subset of users in the m -th user-topic from $U^m = \{u \in \mathbb{U} | p(u | \phi_{u,(m)}) > Q_{75} \forall m \in 1 : K_u\}$, and subset of places in the n -th place-topic from $P^n = \{p \in \mathbb{P} | p(p | \phi_{p,(n)}) > Q_{75} \forall n \in 1 : K_p\}$. Once we partition the groups of users and groups of places, we need to find the top K strongest relationships between the groups of users and groups of places. We can do that by selecting the top K parameters $\theta_{i,j}$ with the largest value. Alternatively, we can select the top K parameters so that we don't need to specify the number of biclusters that we want to extract. We can extract the list of the most relevant biclusters from $\mathcal{B} = \{U^m, P^n | p(z_u = m, z_p = n | \theta, \alpha) > Q_{75} \forall m \in 1 : K_u, \forall n \in 1 : K_p\}$. Where Q_{75} is the 75-th percentile.

4.7 Why does DT2B work?

We show that DT2B is able to extract bicliques by analyzing which records influence the topic assignment of which other records. We discover that the records belonging to the same biclique have the highest mutual influence and so they are most likely to be assigned to the same user-topic and the same place-topic.

4.7.1 Topic influence between a pair of records

We start by rewriting the inference equation (Eq. A.19) in terms of three factors:

$$P(z^u, z^p | u, p, \alpha, \beta) = P(z^u, z^p | \theta) P(u | \phi^u) P(p | \phi^p)$$

The topic influence from the record $\langle u, p \rangle$ to the record $\langle \hat{u}, \hat{p} \rangle$ is given by the following equation:

$$P(z^u, z^p | \hat{u}, \hat{p}, \alpha, \beta) = P(z^u, z^p | u, p, \alpha, \beta) \frac{P(p | \phi^p)}{P(\hat{p} | \phi^p)} \frac{P(u | \phi^u)}{P(\hat{u} | \phi^u)} \quad (4.3)$$

We know that the influence is maximal if $\frac{P(p | \phi^p)}{P(\hat{p} | \phi^p)} \frac{P(u | \phi^u)}{P(\hat{u} | \phi^u)} = 1$ because the topic assignment of $\langle u, p \rangle$ will follow the same equation as the topic assignment for the record $\langle \hat{u}, \hat{p} \rangle$.

If the two records belong to the same biclique, then it must follow that either i) The user in both records is the same ($u = \hat{u}$) ii) The place in both records is the same ($p = \hat{p}$) iii) Trivially, the user and the place is the same in both records.

In these three cases, we can simplify equation 4.3 because either $\frac{P(u | \phi^u)}{P(\hat{u} | \phi^u)} = 1$ or $\frac{P(p | \phi^p)}{P(\hat{p} | \phi^p)} = 1$. It is only in the case when two records are not in the same biclique that equation 4.3 will not be simplified. If equation 4.3 cannot be simplified, both records

will not have a strong influence in the topic assignment of each other. In conclusion, those records that share the same user or the same place have the highest influence in the topic assignment of each other.

4.7.2 Mutual influence for the records in the same biclique

We have proved that a pair of records in the same biclique will have higher influence when compared with a pair of records that do not belong to the same biclique. Therefore, all records in a biclique will mutually reinforce the topic assignment of each other and so the user-topic, place-topic assignment will tend to be the same for all these records. For a better understanding of this fact, we provide a graphical example in Figure 4.6.

First, we represent the visit records in a graph so that each vertex represents the visit of a user to a place (Figure 4.6a). Second, we visualize with a colored set those records that influence the topic assignment of the record $\langle u_1, p_1 \rangle$. In this case, we note that the records $\langle u_2, p_1 \rangle$ and $\langle u_1, p_2 \rangle$ will have higher influence on $\langle u_1, p_1 \rangle$ because those records have either the same place or the same user (Figure 4.6b). Third, we visualize the influence of those records in the biclique composed by $u_1, u_2; p_1, p_2$. We observe how all the records in the biclique mutually influence each other (Figure 4.6c). Fourth, we display an example of two records who do not belong to the same biclique. The records $\langle u_1, p_2 \rangle$ and $\langle u_4, p_4 \rangle$ influence the topic assignment of the record $\langle p_2, u_4 \rangle$. But they will not have a mutual influence (Figure 4.6d). Fifth, we display all the influence sets generated by the records in the dataset. We observe that from the aggregation of all the influence sets, the records including $u_1, u_2; p_1, p_2$ are likely to be assigned to the same user-topic, place-topic pair. We conclude the same for the records including $u_4, u_5; p_3, p_4$ (Figure 4.7).

Fig. 4.6 Demonstration of how DT2B extracts bicliques.

4.7.3 Applications and experimental evaluation

In summary, the advantages of combining topic modeling with biclustering brings the following advantages:

1. Identify the most relevant rows in each bicluster.
2. Identify the most relevant columns in each bicluster.
3. Infer the relevance of the biclusters
4. Extract biclusters under different structures
5. Infer the joint and conditional distributions between the variables
6. The mixture covers the data well. dominant patterns in the data are captured by component distributions

Fig. 4.7 All influences of all pairs in the dataset.

4.7.4 Datasets

Foursquare. The first type of data contains check-in data. Every check-in record can be represented as a $\langle \text{user_id}, \text{place_id}, \text{category} \rangle$ tuple. The records include 227428 check-ins from 1082 users into 38333 venues collected from Apr 2012 to Feb 2013 in New York [110], 573703 check-ins recorded from 2293 users into 61858 venues collected from Apr 2012 to Feb 2013 in Tokyo [110], and 342850 check-ins from 2341 users into 44173 venues collected from Aug 2010 to Jul 2011 in Singapore.

Singtel Cell Location. The second data set contains cell-tower location records which can be represented as a $\langle \text{user_id}, \text{cell_tower lat}, \text{cell_tower lon} \rangle$ tuple. We used 6880128 records from 20830 Singapore's residents and visitors into 4328 cell-towers located across Singapore collected during 2013. This records were provided by Singtel .

NIPS. The third dataset is the text data obtained from papers accepted by NIPS from 1988 until 2003 [80]. The data set contains 40552970 records generated by 2865 authors who wrote 2483 documents using a vocabulary of 14036 words.

MicroArray. The last type of data contains eight different Micro Array gene expression datasets from the Gene Expression Omnibus (GEO). GEO is a public repository designed to be used by many analysis applications. The dataset adheres to academic and industrial standards. Each dataset can be represented by a matrix, where the rows represent the genes and the columns represent samples of DNA strands. Every cell in the matrix corresponds to the regulation of the expression level of a particular gene in a particular sample. The regulation might be positive indicating that the gene is expressed in normal cells. The description and break down of how many genes and samples were used in each of the datasets is presented in Table 4.2.

⁰<http://info.singtel.com/>

Dataset	Genes	Samples	Description
GDS181	12559	84	Human and mouse
GDS589	8799	122	Rat peripheral and brain regions
GDS1027	15866	154	Rat Lung sulphur mustard exposure model
GDS1319	22548	123	C blastomere mutant embryos
GDS1406	12422	87	Mouse brain regions
GDS1490	12422	150	Mouse neural and body tissue
GDS3715	12559	110	Human skeletal muscles
GDS3716	22215	42	Breast epithelia: cancer patients

Table 4.2 Gene expression omnibus datasets.

4.7.5 Biclustering on synthetic data

The inter-topic correlation can be interpreted as the relationship between subsets of users and subsets of places. Semantically, this correlation represents the information of "Which sets of users visit which sets of places", or "Which sets of documents are described by which sets of words". Extracting this relationship corresponds to the same objective of biclustering [49]. Formally, a biclustering algorithm takes as an input a matrix A with dimensionality $\mathbb{I} \times \mathbb{J}$ and it outputs a list of biclusters $\mathcal{B}$ where each bicluster $B_{I,J}$ is defined by a subset of the rows $I \subseteq \mathbb{I}$ and a subset of the columns $J \subseteq \mathbb{J}$. In the user-place space $\mathbb{U} \times \mathbb{P}$, $A_{i,j} = 5$ represents five visits from the i -th user to the j -th location. A bicluster B_{U^m, P^n} is defined by a subset of users $U^m \subseteq U$ and a subset of the places $P^n \subseteq P$.

We want to evaluate DT2B's ability to extract high-frequency biclusters under the structures presented in figures 2.2a, 2.2b, 2.2c, 2.2d because this information represents the solution to the question "Which are the most common groups of users that visit a group of similar places?". Since we do not have ground truth labels for the answers to our question. We propose an experiment where we use three synthetic datasets. The first dataset assumes that every user and every place must belong to exactly one bicluster 2.2a. The second dataset assumes that every user and every place belongs to at least one bicluster 2.2c, and 2.2d. The third dataset assumes that every user must

belong to at most one bicluster 2.2b. We use a normal distribution with a variance σ^2 to represent the visit frequency for a subset of users and a subset of places. All datasets are set to have 100000 users and 100000 places.

As the evaluation metric, we use recovery and relevance [32] as given by $S(\mathcal{B}_1, \mathcal{B}_2) = \frac{1}{|\mathcal{B}_1|} \sum_{B_1 \in \mathcal{B}_1} \max_{B_2 \in \mathcal{B}_2} |B_1 \cap B_2|$ Where $\mathcal{E}$ is the list of expected biclusters. Relevance is defined as $S(\mathcal{B}, \mathcal{E})$ and Recovery is defined as $S(\mathcal{E}, \mathcal{B})$. Intuitively, these metrics are an analogy to Recall and Precision.

We compare results with bicluster algorithms that best performed on tested data with varying conditions, including varying noise and varying numbers of biclusters [32],[70]. The implementation of all algorithms were taken from the author's websites.

In Figure 4.9, we display the results of this experiment. In the first dataset, we observe an improvement of the spectral bipartite graph over other models, that is because spectral bipartite graph assumes that all places and all users must be assigned to exactly one bicluster and so the algorithm is fitted for the assumptions of the first dataset, when we relax the assumption of dataset 1 in datasets 2 and 3, the results of the spectral bipartite graph decrease when compared against DT2B, overall DT2B yields better results when initialized using DT2B.

The spectral bipartite graph [28] algorithm performs much better than other baseline models because the spectral bipartite graph algorithm models the relation between users and places with a bipartite graph weighted with the number of visits. Therefore, this algorithm is designed for extracting high-frequency biclusters. Other biclustering algorithms aim to find similarities between rows and columns without taking into account the absolute value of their relationship.

In addition, we visualize one sample of the biclusters extracted by DT2B to compare the difference between initializing DT2B using two LDA models and DT2B using two K-means algorithms. The visualization is shown in Figure 4.8. We use one color to

Fig. 4.8 Recovery and relevance results for bicluster classification. The datasets are visualized with a matrix whose color represent the number of visits of users (rows) to places (columns).

represent one bicluster. On the right side of the figure, we can see the input dataset where the number of visits from user i -th to place j -th is given by a value between 0 and 200.

As a result, we observe that initializing DT2B with two independent LDAs (LDA-DT2B) is a better way to find more relevant biclusters for two reasons. First, the number of extracted biclusters is more precise since LDA-DT2B adds more weight to the topic-correlation that represents the ground truth biclusters, i.e., LDA-DT2B has a better representation of the topic inter-correlation. Second, LDA-DT2B extracts biclusters that better fit the shape of the ground truth biclusters. This is because LDA-DT2B extracts topics that better represent the subsets of users and subset of places that we expect as ground truth.

4.7.6 Biclustering in microarray data

Biclustering is most commonly applied to microarray data. Many types of biclusters have been found to coexist in microarray datasets [82]. Biclusters are an association

Fig. 4.9 Different initialization methods for extracting biclusters with DT2B

between groups of genes with samples or disease phenotypes. To validate the extracted biclusters, we use Gene Ontology Enrichment [34]. In this experiment, we show that DT2B can successfully mine heavy biclusters from DNA microarray data.

We follow the evaluation procedure proposed by Eren et al [32]. That is, we extract biclusters from microarray data, and then we perform gene ontology enrichment analysis over the extracted biclusters. A bicluster is considered to be enriched if at least one term from the Biological Process Gene Ontology was enriched at the $P = 0.05$ level after multiple test correction. Evaluation is carried out by extracting the proportions of the enriched biclusters to the extracted biclusters aggregated for the **eight microarray datasets**. The proportion of enriched biclusters is a widely used method to compare different biclustering methods [39, 32].

For DT2B, we set the number of both row-topics (gene-topics) and column-topics (sample-topics) to 15. The data must be pre-processed since microarray data contains real values and our model is designed for discrete positive values. We propose to analyze the positive values and the negative values separately. First, we replaced the negative values with zero so that we extract those genes and samples where there is a high expression of normal cells relative to tumor cells. Then, we replaced the positive values with zero so that we extract those genes and samples where there is a high expression of tumor cells relative to normal cells. Finally, we discretize the data so that we can apply our model.

Fig. 4.10 Proportion of the enriched biclusters for DT2B. The results are compared with the biclustering algorithms studied by Eren et al on five different significance level α [32].

Figure 4.10 shows the proportions of the enriched biclusters to the extracted biclusters aggregated for the eight datasets. We also show the effect of increasing the significance level α . Note that in this section, α does not refer to a hyperparameter. Results suggest that the heavy bicluster is a bicluster type worth studying in microarray data. The proportion of enriched biclusters of DT2B is higher than those of other methods. This is particularly significant when the α level is selected to be greater than 0.1%. Table 4.3 shows the terms associated for the biclusters with the lowest p-value for the GDS581 dataset. DT2B revealed new terms not discovered by other bicluster algorithms. This is because DT2B targets the extraction of heavy biclusters with a flexible structure, while most algorithms use specific and rigid structures. The number of genes associated with each bicluster is higher than other algorithms while maintaining low p-values. This indicates that DT2B is a complementary model capable of extracting biclusters that may not be easy to extract with other algorithms.

Rows,cols	Terms	P-value
502,20	Hydrogen ion trans-membrane transport	2.52e-17
566,4	ATP metabolic process	1.17e-15
674,18	Cellular amide metabolic process	2.72e-15
685,20	Hydrogen ion trans-membrane transport	4.21e-14
643,10	Peptide metabolic process	4.17e-11

Table 4.3 Five most enriched terms on GDS589.

4.7.7 Reviewer recommendation.

We evaluate DT2B in an application where we require to make use of the Dual LDA. As suggested by Rosen-Zvi et al, automated reviewer recommendations is an application that requires extracting author similarity [87]. We can capture this similarity by using an LDA so that we obtain author-topics. We propose a modified version of the application where we generate a list of likely authors who may have written a document given the collection of words in a document. Our assumption is that if we can correctly identify the authors of a document, then we should be able to identify similar authors who may have written the document and hence they could serve as reviewers for the document. Since the application doesn't require any information linking the document to the authors, we could recommend reviewers to documents with a single author or documents written by an author not present in the corpus.

More formally, given a document $\mathcal{D}$ represented by a set of words w , we can use DT2B to extract the probability of an author a to write a word w from equation 4.4

$$P(a, w) = \sum_{z_a} \sum_{z_w} P(a | \Phi_{z_w}, \beta_w) P(a | \Phi_{z_a}, \beta_a) P(z_a, z_w | \Theta) \quad (4.4)$$

Fig. 4.11 Reviewer recommendation

Then, we use $\prod_{w \in \mathcal{D}} P(a|w)$ to find the probability of an author to have written the document $\mathcal{D}$. For comparative purposes, we also extract $P(a|w)$ using an LDA by representing the document level using words and the word-level using the set of all authors in the corpus. i.e., we use an LDA to represent a word as the set of authors who have written this word. In this model, the probability of an author a to write a word w is calculated from equation 4.5.

$$P(a|w) = \sum_{z_a} P(a|\Phi_{z_a})P(z_a|\Theta_w) \quad (4.5)$$

We use a similar formulation to estimate $P(a|w)$ using the Correlated Topic Model (CTM) [60].

For this experiment, we split the papers in the NIPS corpus into a training set and a testing set, after deleting all papers in the test set with one author and those papers where the authors are not in the training set, we obtain as the test set 525 papers with two authors, 305 with three authors and 111 with four authors. We evaluate the precision and recall at n by selecting the top n authors from $P(a|\mathcal{D})$. In all the models

Trial	Uniform			K-means		
	LDA	DT2B	LDA+DT2B	LDA	DT2B	LDA+DT2B
1	0.032	0.032	<u>0.016</u>	0.104	0.024	<u>0.008</u>
2	0.032	0.272*	<u>0.016</u>	<u>0.296*</u>	0.304*	<u>0.296*</u>
3	<u>0.008</u>	0.272*	<u>0.008</u>	0.208	0.336	<u>0.000</u>
4	<u>0.016</u>	0.016	<u>0.016</u>	0.448	0.416	<u>0.208</u>
5	0.040	0.024	<u>0.016</u>	0.224	0.112	<u>0.008</u>
6	0.312*	0.288*	<u>0.264*</u>	0.288*	<u>0.144</u>	0.272*
7	0.176	0.016	<u>0.008</u>	0.112	0.128	<u>0.048</u>
8	<u>0.000</u>	0.152	<u>0.000</u>	<u>0.016</u>	<u>0.016</u>	<u>0.016</u>
9	0.304*	<u>0.272*</u>	0.304	0.024	0.016	<u>0.008</u>
10	0.184	<u>0.000</u>	0.016	0.344*	0.304*	<u>0.160</u>
Average	0.1104	0.1344	0.0664	0.2064	0.1800	0.1024

Table 4.4 Average misclassification error for ten random initializations. The initialization method is specified in the headers of the table. The asterisk represents those cases where the models completely confused two topics.

we set the number of topics equal to fifty. In the case of DT2B both author-topics and word-topics are set to fifty as well.

We present the results in figure 4.11. We observe an improvement of DT2B over CTM and an improvement of CTM over LDA. This is expected as both CTM and DT2B generalize LDA. Overall we can see how papers with larger number of authors have a better performance when trying to identify the authors of the paper. As expected, the precision decreases as we increase the list of recommended authors and in contrast, the recall increases. Finally, we note that initializing GLDA with a uniform distribution leads to better performance than initializing it with the result of an LDA and its dual LDA, though the convergence time is about an order of magnitude larger.

4.7.8 User and place classification

DT2B can also be used to classify data using the latent topics in the scenario where the classes represent the expected clusters. In a hard classification, we can assign class

z to user u_i using $\hat{z}_u = \arg \max_{z_u} P(u = u_i | \Phi_u, z_u)$ by using the extracted topics. In a soft-classification mode, one could extract $P(z_u | u = u_i) = \frac{P(u = u_i | \Phi_{u, z_u}) P(\Phi_{u, z_u})}{P(u = u_i)}$ where $P(u = u_i)$ can be found by counting from the given data and $P(\Phi_{u, z_u}) = \frac{\prod_{i=1}^K \Gamma(\beta_i)}{\Gamma(\sum_{i=1}^K \beta_i)} \prod_{i=1}^K \Phi_{u, i}^{\beta_i - 1}$ as specified by the Dirichlet prior with parameter β

In this experiment we test DT2B's ability to represent individual user-topics and individual place-topics. We selectively filter the cell-tower data so that we can obtain clear ground truth topics of users and places. We select all the visit records of five groups of 250 users whose home and work correspond to the same cell tower location. We assume as ground truth that there exists five user-topics, where each topic represents a group of 250 users. For every user u , we assign the user-topic $\hat{z}_u$ using $\hat{z}_u = \arg \max_{z_u} P(u = u_i | \Phi_u, z_u)$. For the evaluation, we compare the average misclassification error of DT2B against an LDA. We initialize the models uniformly and using K-Means. In addition, we also use the results of an LDA and its dual LDA to initialize the DT2B.

Table 4.4 shows a consistent improvement of DT2B model when initialized using K-means and the independent LDA models. However, in those cases where K-means initialization is not so convenient such in trials 7 and 8, we cannot expect DT2B to solve the confusion between the extracted topics of the independent LDA models. We also observe that LDA is a better way to initialize DT2B model. We attribute the improvement of DT2B over LDA to the fact that DT2B models an iterative inference of both user-based models and location-based model. In contrast, the LDA is a location-based model.

Additionally, we used the New York Foursquare's dataset to compare the extracted place-topics of LDA and DT2B when initialized with the results of independent LDA models. In Figure 4.12 we point out all differences that we consider to be an improvement. Typically, DT2B is able to solve some confusion between the places

Fig. 4.12 Comparison for place-topic extraction between LDA and DT2B. DT2B was initialized with the results of the LDA. The numbers represent the number of samples allocated to a topic

represented by two topics. For example, after convergence of the LDA model, *Gym* appeared in topic 1 and 3. However, when we used DT2B to improve the convergence of LDA, *Gym* was moved to topic 3 where it is better represented. Our model also improved the ranking of the places inside a topic. For example, in topic 5, DT2B allocated more instances to *Train Station* in a topic that clearly corresponds to places related to transportation.

4.7.9 Dimensionality reduction

Topic models are commonly used for dimensionality reduction [13]. The new dimensionality can be reduced by using the inferred topics. In the LDA model, for the user-place domain, every user will be assigned a mixture of topics and every topic will correspond to a mixture over the set of places. i.e., we are describing $\hat{p}(P|U)$ in a lower dimension. An optimal representation of the lower dimension conditional distribution would be able to get close to the truth distribution $p(P|U)$. We assume that P and U are multinomial random variables representing the set of places and users respectively.

In this experiment, we assume the ground truth to be the maximum likelihood estimation of $p(P|U)$. For the evaluation, we extract the estimated value of $\hat{P}_{lda}(P|U; z)$ using the LDA model as $P_{lda}(U|P) = \sum_{z_u}^{Ku} p(z_u|\Theta, \alpha)p(U|\Phi_{z_u}\beta_u)$. Similarly using the dual LDA Formulation, we extract $p(U|P; z)$. For the DT2B model we approximate the conditional distribution by first finding the joint distribution $p(U, P; z_u, z_p) = \sum_{z_u}^{Ku} \sum_{z_p}^{Kp} p(z_u, z_p|\Theta, \alpha)p(U|\Phi_{z_u}\beta_u)p(P|\Phi_{z_p}\beta_p)$ and then we extract the conditional distributions by normalizing the joint distribution. We use the Bachattaryya distance to measure the distance between distributions. In the case of multinomial distributions the Bachattaryya distance between p and q corresponds to $Battacharyya(p, q) = -\ln \left(\sum_x \sqrt{p(x), q(x)} \right)$. We report results based on averaging the results of running the experiment 50 times.

K	$P(P U)$			$P(U P)$		
	DT2B	LDA	Improvement	DT2B	LDA	Improvement
3	.261 (.003)	.262 (.005)	44.7%	.644 (.006)	.647 (.005)	58.2%
6	.227 (.002)	.245 (.006)	76.3%	.592 (.006)	.621 (.005)	69.9%
9	.214 (.003)	.242 (.005)	78.1%	.565 (.007)	.623 (.005)	89.4%
12	.204 (.004)	.559 (.008)	99.7%	.550 (.007)	1.00 (.004)	99.8%
15	.193 (.004)	.528 (.011)	99.9%	.523 (.006)	1.01 (.005)	99.9%

Table 4.5 On the left side of the table, we show the average Battacharyya distance between the approximated distributions $\hat{P}_{LDA}(P|U)$, $\hat{P}_{DT2B}(P|U)$ and the estimated distribution $P(P|U)$. We assume the maximum likelihood estimation of $P(P|U)$ as ground truth. On the right side, we display the Battacharyya distance between the approximated distributions $\hat{P}_{LDA}(U|P)$, $\hat{P}_{DT2B}(U|P)$ and the estimated distribution $P(U|P)$. We assume the maximum likelihood estimation of $P(U|P)$ as ground truth.

Fig. 4.13 Results of Table 4.5 for every user, we sort the users and places by their average Battacharyya distance to facilitate the visualization of the results.

In Table 4.5, we present the average Battacharyya distance between the approximated distributions $\hat{P}_{LDA}(P|U)$, $\hat{P}_{DT2B}(P|U)$ and the estimated ground truth distribution $P(P|U)$. We also show the percentage of the users whose average distance to the ground truth distribution is improved as well as the percentage of places improved by the DT2B model.

DT2B improves the place-distribution for around 75% of all the users and 70% of the user-distributions per place. These results hold independently of the selected number of topics. However, when we use a small number of topics, the performance of DT2B and LDA tend to converge. Naturally, as we increase the number of topics, we obtain smaller distances to the truth distributions. When we surpass 10 topics, LDA decreases its performance dramatically. When we use LDA to extract larger number of topics, several topics end up being repeated. This is a consequence of LDA modeling the correlation of user to topics without considering the relationship between users. In Figure 4.13, we visualize the distance of the place-distribution to the ground truth for every individual and the distance of the user-distribution to the ground truth for every place. The figure shows how changing the number of topics reflects an homogeneous improvement or decrease of performance for all the users.

4.7.10 Dual topics visualization

Finally, we created a visualization to show the extracted topics and their correlation. We tried this visualization on two domains. Firstly, we used a Movie rating dataset using one million ratings from 6000 users on 4000 movies¹. We selected this dataset since we also have available user demographic information that can be used to explain the user topics. We applied DT2B to the user-movie domain in order to extract *Which groups of users watch which types of movies*. The results are displayed in Figure

¹<http://grouplens.org/datasets/movielens/>

Fig. 4.14 On top, we show four movie-topics represented by a) Most representative movies and its year of release. b) Histogram of the movie genres weighted by the number of instances that a movie was allocated to a topic. On the middle we represent the correlation between topics using a parallel sets visualization. On the bottom, we show the user-topics represented by a) Histogram of the user's profession b) Histogram of the user's age. Both histograms are weighted by the number of instances that a user was allocated to a topic.

4.14. We use the parallel sets visualization to display the correlation of topics ². In this experiment, we obtained a one to one relation between the user-topics and the movie-topics. The label of topics is based on our judgments. Our main findings indicate that users related to academia have higher preference towards drama and comedy movies, while users related to the fields of business and engineering have a higher interest in action movies. As expected, older users naturally prefer to watch older movies.

Secondly, we tried this visualization on the NIPS dataset. For the data processing, we filtered those words that don't exhibit a particular meaning. E.g., data, figure and image. The results are displayed in Figure 4.15. In this experiment, we extracted six author topics and ten word topics to be able to compare with NIPS technical ³ which correspond to: algorithms and architectures , applications, brain imaging, cognitive science and artificial intelligence, control and reinforcement learning, hardware technologies , learning theory, neuroscience, speech and signal processing , visual processing. From the visualization, we observe high correlation between the extracted word topics and NIPS technical areas as the fields of 'hardware technologies', 'learning theory', 'neuroscience' and 'speech recognition'. Different from the expected NIPS technical areas, we find a topic categorization that falls into solving approaches rather than technical areas. For instance, probability, optimization and artificial neural networks emerge each as topics, while in NIPS technical areas, they are all considered to be part of the algorithms and architectures technical area.

²<https://nips.cc/Conferences/2008/CallForPapers>

³<https://www.jasondavies.com/parallel-sets/>

Fig. 4.15 Author-topics, word-topics and its interrelation. The width of the box surrounding each topic represents the total amount of samples allocated to the topic.

4.8 Summary

In this chapter, we presented the first contribution of our thesis which corresponds to the design and implementation of a novel topic model for the extraction of biclusters. We started our explanation by introducing the limitations of existent biclustering algorithms. Then, we presented how topic models are equipped with the necessary characteristics to reduce some of the limitations of existent biclustering algorithms. However, one challenge of applying topic models to extract biclusters is that existent topic models are designed to cluster one random variable, but they cannot extract simultaneously two random variables. As a contribution, we built a novel probability distribution that empowers topic models with the ability to extract biclusters and thus, our model relaxes some of the assumptions of existent biclustering algorithms. We showed the utility of our model to applications using text data, microarray data and location data. In addition, we demonstrated that our model generalizes latent Dirichlet allocation model as well as the author-topic model. Finally, we presented a mathematical intuition for why our proposed model is able to extract biclusters in bivariate data.

Chapter 5

Inference of Topic Models

In this chapter, we focus on the learning of topic models with the objective of improving the computational complexity as well as the predictive power of current inference algorithms.

Firstly, in Section 5.1, we empower the collapsed Gibbs sampling (CGS) algorithm with the use of soft-topic assignments as opposed to the standard one-topic assignment. The novel algorithm learns by continuously exploring the whole topic space instead of the single dimension specified by the one-topic assignment and as a result, the novel method achieves a higher predictive power than CGS. The transformation of the latent topic also enables the learning to be solved via substitution sampling which can be seen as a recursive compounding of distributions. Hence, we label the method as recursively compound allocation (RCA). As an additional advantage, RCA is deterministic which facilitates the hard to assess convergence of the sampling methods [15]. In addition, we also present a link that relates the mean-field variational inference to our approach. To the best of our knowledge, our method is the first one to study a deterministic approach within Gibbs sampling, and it is also the first approach to find a connection between the mean-field variational field and a Gibbs-based inference method.

Secondly, in Section 5.2, we study an state augmentation method which claims a more accurate inference for topic models [117]. The advantage of this method relies on augmenting the number of latent variables for every word in the dataset. Naturally, we propose to maximize this advantage and mathematically we encounter that the state augmentation method with maximal latent state replication corresponds to our previously presented RCA method.

We go beyond high predictive power and we improve RCA’s inference after analyzing how the convergence of Algorithm 4 yields another representation based on a system of equations. The system of equations can be solved using a fixed-point iterator. We label this method as Heron inference ¹. Perhaps the major benefit of Heron inference is that fixed-point methods have been well studied to solve systems of equations, and their iterative solvers are distributable [8]. The connection to a fixed-point system may unleash some powerful tools on the acceleration of iterative solvers [100].

Heron inference can be distributed into as many cores as data points without affecting the predictive power. Typically, distributed algorithms within the Gibbs family affect the predictive power of the inference methods as the number of cores grows larger. This increases the complexity of GPU implementations. Because our proposed method does not use sampling, it avoids complex pseudo- and quasi-random number generator routines provided by GPU libraries. As result, Heron relies on simpler and more efficient GPU implementations. Finally, we reduce the running time of Heron by proving that it is not required to sample repeated document-word pairs such as those pairs with repeated words within a document.

¹We coin the term Heron inference in honor of Heron of Alexandria. Heron is perhaps thought to be the first person who solved iteratively a fixed-point function with the objective of computing the square root of a number

5.1 Recursively Compound Allocation

5.1.1 Principle of recursive compounding

In LDA inferred by CGS, we compute the posterior distribution for an observed document-word pair by using the topic assignments of all the other pairs. Then, we rely on the posterior distribution to sample a single topic label for the pair. As argued earlier, by allowing the soft-topic assignment, i.e., directly employing the distributions of topic assignments instead of exact assignments of latent topics, we may relax the inference process and enhance the predictive power of LDA.

Generally, a distribution whose parameter is again distributed according to another probability distribution is often known as a compound probability distribution. Then, in order to improve the predictive ability of CGS, for each given document-word pair, instead of sampling the exact topic label, we propose to use the compound distribution as the soft-topic assignment. In other words, the posterior topic distribution for the given document-word pair will be computed directly from the distributions of topic assignments of all the other document-word pairs. In this way, we develop a new deterministic inference method within the Gibbs sampling framework, named recursively compound allocation (RCA).

First, we illustrate how to transform Gibbs sampling into a deterministic process of compounding distributions using a simplified task.

From random sampling into a deterministic process

Given a set of random variables $X_1, X_2, \dots, X_M$, we are typically interested in estimating densities related to these random variables. E.g., the marginal densities $f(X_1), f(X_2), \dots, f(X_M)$. However, it is often the case that these distributions cannot

Fig. 5.1 Using the function $g()$ to transform an r.v. Z_j with categorical distribution to an r.v. Y_j with degenerate Dirichlet distribution.

be found analytically. Hence, we resort to iterative methods, such as Gibbs sampling, to estimate the marginal distributions.

Without loss of generality, we consider two random variables X_1 and X_2 . As a requirement of Gibbs sampling, we assume that we know the analytic form of the conditional densities $f(X_1|X_2)$ and $f(X_2|X_1)$. Then, Gibbs sampling performs the following procedure iteratively:

1. Select an initial sample x_2^0
2. Sample x_1^1 from $f(X_1|X_2 = x_2^0)$
3. Sample x_2^1 from $f(X_2|X_1 = x_1^1)$
4. Repeat until convergence

As the number of iterations increases, the samples x_1 and x_2 approximate to the samples from the joint distribution $f(X_1, X_2)$. Unfortunately, the exact number of iterations to convergence is unknown, since it is not possible to decide the convergence of a distribution using the samples.

We would transform CGS into the deterministic RCA, if in step 2 we do not sample x_1^1 from $f(X_1|X_2 = x_2^0)$, but instead, we define a random variable X_1^1 distributed according to the density $f(X_1|X_2^0)$, where X_2^0 is an initial distribution assigned to X_2 . Then, we use the newly defined random variable X_1^1 as the conditional variable to obtain the density of X_2^1 . Formally, X_1^1 and X_2^1 are compound distributions, as

their parameters depend on random variables, and not on sample assignments. To summarize,

1. Initialize the parameters of the density for X_2^0
2. Define $X_1^1 \sim f(X_1|X_2^0)$
3. Define $X_2^1 \sim f(X_2|X_1^1)$
4. Reiterate until convergence.

To obtain the density of X_1^1 , we apply the principle of compound distribution [19] as follows:

$$f(X_1^1) = \sum_{x_2 \in R_{X_2^0}} f(X_1|X_2^0 = x_2)f(X_2^0 = x_2), \quad (5.1)$$

where $R_{X_2^0}$ denotes the support of X_2^0 . We can then compute $f(X_2^1)$ in a similar way.

Although it may seem as if the new deterministic process, which leverages the compound distributions of topic assignments instead of particular topic labels, converges to a different result, Gelfand and Smith showed that this deterministic process, also known as the substitution algorithm, converges to the desired marginal distributions of Gibbs sampling [40]. Despite the subtlety, by transforming Gibbs sampling into a deterministic process, we obtain an additional benefit of having the ability to flag the convergence status. The deterministic procedure is able to measure the distance of the assigned distributions between two given iterations. When this distance is smaller than a predetermined value, the method is able to specify the convergence of the inference.

Substitution sampling is intractable for LDA and current inference methods still rely on random sampling. However, we now show how we can transform the latent random variable of a topic model so that substitution sampling can be applied.

Latent Dirichlet allocation inferred via RCA

Following the aforementioned steps, in this section, we transform the inference of LDA via collapsed Gibbs sampling (LDA-CGS) into a deterministic process.

First, we assign a topic distribution to the j^{th} document-word pair given random variable distributed as $\hat{Z}_j \sim p(Z_j|Z_{-j}, \mathbf{X})$, where Z_{-j} corresponds to topic distributions of all the document-word pairs sans the j^{th} pair. Since the conditionals Z_{-j} are also random variables, Z_j is distributed according to the following compound distribution:

$$f(Z_j|Z_{-j}, X) = \sum_{Z_{-j}} f(Z_j)f(Z_1, \dots Z_{\neq j}, \dots Z_M) \quad (5.2)$$

Similarly, the method applies the same equations to all other document-word pairs until their posterior distributions converge to the target distributions $P(Z_j)$ of collapsed Gibbs sampling.

Unfortunately, Equation 5.2 is intractable, as $f(Z_1, \dots Z_{\neq j}, \dots Z_M)$ is analytically unknown. We can deal with this issue if we define a function $g()$ that transforms each categorical random variable Z_j into a *degenerate Dirichlet* distributed random variable, i.e., $Y_j = g(Z_j)$.

Figure 5.1 illustrates this transformation using as example a random variable Z with three possible outcomes $R_X \in \{1, 2, 3\}$. The probability is assigned to each outcome corresponding to η_1, η_2 , and η_3 , respectively. The mapping $g()$ transforms Z into a random variable Y distributed according to degenerate Dirichlet distribution with parameters η , where $\eta = [\eta_1, \eta_2, \eta_3]$. The degenerate Dirichlet distribution is a special case of the Dirichlet distribution, in which all its probability mass is located at the only possible outcome η . The parameters of the degenerate Dirichlet distribution correspond to $\lim_{\alpha \rightarrow \infty} Dirichlet(\alpha\eta)$. In summary:

$$Y = g(Z) \sim deg-Dir(\eta) \text{ if } Z \sim Categorical(\eta).$$

We apply the transformation to each of the random variables Z_j as follows:

$$Y_j \sim g(Z_j) \implies f(Y_j) = g(f(Z_j)) = f(g(Z_j))$$

$$f(Y_j) = \int_{\eta_{-j} \in R_{Y_{-j}}} f(Y_j | Y_{-j} = \eta_{-j}) f(Y_{-j} = \eta_{-j}) dR_{Y_{-j}} \quad (5.3)$$

$$= f(Y_j | Y_{-j} = \eta_{-j}) \quad (5.4)$$

Under the transformation, the summations in Equation 5.2 become a set of integrals, as we are now compounding continuous distributions. In addition, we note that the density of Y_{-j} will *only* be equal to one when it is evaluated at η_{-j} , i.e., $f(Y_{-j} = \eta_{-j}) = 1$. Hence, Equation 6 *only* has values different than zero when $Y_{-j} = \eta_{-j}$. This leads to a considerable simplification. The solution then becomes tractable, as presented in Equation 5.4.

As a result, the substitution algorithm in conjunction with the mapping $g()$ overrides $D_{x_j^{(d)}}$, $W_{x_j^{(w)}}$, and N_j of LDA-CGS (Equations 3.8, 3.9, and 3.10) as follows:

$$D_{x_j^{(d)}} = \sum_{m=1, m \neq j}^M \mathbb{1}(x_j^{(d)} = x_m^{(d)}) * \eta_m \quad ; \quad N_j = \sum_{m=1, m \neq j}^M \eta_m W_{x_j^{(w)}} = \sum_{m=1, m \neq j}^M \mathbb{1}(x_j^{(w)} = x_m^{(w)}) * \eta_m$$

where η_m corresponds to the realization of the random variable Y_m . Upon convergence, the parameters are estimated from the following equations:

$$\hat{\theta}_{x_j^{(d)}} \propto (D_{x_j^{(d)}} + \alpha) \quad ; \quad \hat{\phi}_{x_j^{(w)}} \propto (W_{x_j^{(w)}} + \beta) \quad (5.5)$$

Formally, we name this process as recursively compound allocation (RCA). Algorithm 4 summarizes the main steps of applying RCA to inference of LDA.

As shown in Equations 3.8, 3.9, 3.10 and 5.1.1, the relationship between CGS and RCA is intuitive. It is clear that the calculation for the posterior density has

transformed the caching for samples z_j from the posterior distribution into the caching of the parameters η_j for the posterior distribution. The interpretation is that RCA assigns a mixture distribution over topics to each observed document-word pair, while in CGS the distribution is used to sample a single topic label. As a consequence, RCA provides a more flexible learning strategy for topic assignment compared to CGS.

During inference process, the function $g()$ has transformed the vanilla LDA with categorical latent variables into a tractable model with degenerate Dirichlet latent variables. Because Dirichlet distributions sample over a probability simplex, this transformation is responsible for empowering LDA with soft-topic assignments. In this way, the posterior of RCA is clearly different from the posterior of CGS. Thus, the new method RCA could perhaps lead to improved predictive performance for the inference of LDA compared to CGS. This justifies the differences in predictive perplexity in the experiments (See Section 6.2).

Applying RCA to dual topic models

We note that, in addition to latent Dirichlet allocation (LDA), RCA can be also applied to inference of other probabilistic topic models, such as dual topic model (DT2B) [90], relational topic model [22], and author-topic model [87], etc. In contrast to many other topic models, DT2B models the pair-wise dependency between two types of latent

Algorithm 4 Recursively Compound Allocation for LDA

- 1: **Input** Hyperparameters K, α, β , dataset $\mathbf{X}$, convergence threshold ϵ
 - 2: Initialize $f(Y_j|Y_{-j})$, $\forall j \in 1 \dots M$, and set initial iteration $i = 1$
 - 3: **while** marginal densities have not converged **do**
 - 4: **for** $j \in 1 \dots M$ **do**
 - 5: $Y_j^i \sim f(Y_j^i|Y_{<j}^i, Y_{>j}^{i-1})$ from Equation 5.4
 - 6: **end for**
 - 7: Converge if $\text{distance}(f(Y_j^i), f(Y_j^{i-1})) \leq \epsilon$
 - 8: $i = i + 1$
 - 9: **end while**
 - 10: Estimate $\hat{\theta}$ and $\hat{\phi}$ using Equation 5.5. **return** $\hat{\theta}, \hat{\phi}$
-

topics [90]. In this section, we turn our attention to applying RCA to DT2B, and may shed some light on the inference procedure of such kind of topic models.

DT2B takes as input a co-occurrence data matrix that underlies the inter-relationship between *row* and *column* variables. E.g., user-location matrix where each entry means the number of times a user (row) visited a location (column). Then, it extracts the row-topics Z^r or distribution over the rows of the co-occurrence matrix, the column-topics Z^c or distribution over the columns of the matrix, and the joint distribution of both column and row topics. This joint distribution models the dependency between the two types of topics. Therefore, we cannot transform Z^r or Z^c independently into Y^r or Y^c as done in LDA.

To apply RCA to DT2B, we first derive the conditional posterior of CGS for DT2B in Equation 5.6:

$$f(Z_j^r, Z_j^c | Z_{-j}^r = z_{-j}^r, Z_{-j}^c = z_{-j}^c) \propto \frac{(D_{x_j^{(d)}} + \beta^r) (W_{x_j^{(w)}} + \beta^c)}{(N_j^r + R\beta^r) (N_j^c + C\beta^c)} (P_j + \alpha) \quad (5.6)$$

where R and C are the numbers of rows and columns in the input matrix respectively, and the $D_{x_j^{(d)}}$, $W_{x_j^{(w)}}$, P_j and N_j^r are count vectors computed using the indicator function

$\mathbb{1}()$:

$$\begin{aligned}
D_{x_j^{(d)}} &= \sum_{m=1, m \neq j}^M \mathbb{1}(x_j^{(d)} = x_m^{(d)}) * z_m^r \\
W_{x_j^{(w)}} &= \sum_{m=1, m \neq j}^M \mathbb{1}(x_j^{(w)} = x_m^{(w)}) * z_m^c \\
P_j &= \sum_{m=1, m \neq j}^M z_m^r \cdot z_m^c \\
N_j^r &= \sum_{m=1, m \neq j}^M z_m^r ; \quad N_j^c = \sum_{m=1, m \neq j}^M z_m^c
\end{aligned} \tag{5.7}$$

We need to maintain the dependency between the two types of topics when mapping the joint distribution of the random variables Z^r and Z^c to the random variables Y^r and Y^c . This is shown in Equation 5.8

$$\begin{aligned}
Y_j^r, Y_j^c \sim g(Z_j^r, Z_j^c) &\implies f(Y_j^r, Y_j^c) = f(g(Z_j^r, Z_j^c)) = \\
&\int_{\eta_{-j} \in R_{Y_{-j}^r, Y_{-j}^c}} f(Y_j^r, Y_j^c | Y_{-j}^r, Y_{-j}^c = \eta_{-j}) f(Y_{-j}^r, Y_{-j}^c = \eta_{-j}) dR_{Y_{-j}^r, Y_{-j}^c} \\
&= f(Y_j^r, Y_j^c | Y_{-j}^r, Y_{-j}^c = \eta_{-j})
\end{aligned} \tag{5.8}$$

Note that $\boldsymbol{\eta}$ does no longer correspond to the parameters of a degenerate Dirichlet distribution as in the case of LDA shown in Equation 5.4. Instead, $\boldsymbol{\eta}$ corresponds to a bivariate degenerate Dirichlet distribution. To obtain the marginal density of Y^r and Y^c , we can integrate $\boldsymbol{\eta}$ as shown in Equation 5.9.

$$\eta_j^r = \sum_{k_c}^{K_c} \boldsymbol{\eta}_j \quad ; \quad \eta_j^c = \sum_{k_r}^{K_r} \boldsymbol{\eta}_j \tag{5.9}$$

As a result, the posterior equation of DT2B-RCA replaces the values of z^r presented in Equations 3.8, 3.9 and 3.10 for η^r . Similarly, RCA replaces the values of z^c for η^c for the factors $D_{x_j^{(d)}}$, $W_{x_j^{(w)}}$, N_j^r and N_j^c . The factor P_j depends on the joint assignment of Y_j^r and Y_j^c . Therefore, the factor P is computed as:

$$P_j = \sum_{m=1, m \neq j}^M \boldsymbol{\eta}_m \quad (5.10)$$

The flexibility in topic assignment by RCA results in improved predictive performance, as shown empirically in Section 5.4. Moreover, the additional key benefits of RCA are threefold: 1) RCA has a guaranteed convergence; 2) It is allowed to assess the status of convergence; and 3) In RCA, we can directly compute the marginal distributions $p(Y_j)$ and the parameters of the topic models

5.1.2 Proof for convergence of RCA

Without much loss of generality, it is sufficient to show the equivalence between RCA and Gibbs sampling using the case of two joint Bernoulli random variables X and Y . We note that the proof for the convergence of Gibbs sampling in [19] directly leverages compounded distributions rather than sampling. In the paper, they showed that the desired marginal distribution $p(X)$ corresponds to the stationary distribution of a Markov chain with transition matrix $A_{X|X} = A_{Y|X}A_{X|Y}$, where $A_{Y|X}$ and $A_{X|Y}$ are the conditional densities:

$$A_{Y|X} = f_{Y|X}(y) ; \quad A_{X|Y} = f_{X|Y}(x)$$

Therefore, the approximated density of the marginal distribution at the $i^{th} + 1$ iteration can be found from the approximated marginal distribution:

$$f_{X^{i+1}}(x) = f_{X^i}(x)f_{Y^i|X^i}(y)f_{X^i|Y^i}(x) \quad (5.11)$$

As we run for more iterations, the approximated marginal distributions approach the target marginal distributions, i.e., $\lim_{i \rightarrow \infty} f_{X^i}(x) \rightarrow f_X(x)$. The connection with RCA follows, as it can be observed that the intermediary steps of the Equation 5.11 correspond to the compounding process. This can be checked by proposing that the combination of the first two factors corresponds to the approximated marginal $f_{Y^i}(y)$.

$$f_{X^{i+1}}(x) = f_{Y^i}(y)f_{X^i|Y^i}(x) \quad (5.12)$$

$$f_{Y^i}(y) = f_{X^i}(x)f_{Y^i|X^i}(y) \quad (5.13)$$

Following the same rule, we propose $f_{X^i}(x) = f_{Y^{i-1}}(x)f_{X^{i-1}|Y^{i-1}}(y)$. We show that the proposition about $f_{Y^i}(y)$ is correct by plugging the proposition of $f_{X^i}(x)$ into the above equation. Then we end up with:

$$f_{Y^i}(y) = f_{Y^{i-1}}(x)f_{X^{i-1}|Y^{i-1}}(y)f_{Y^i|X^i}(y)$$

This corresponds to the Markov chain equation whose stationary distribution targets the marginal distribution of $f_Y(y)$. Thus, the proposition for $f_{Y^i}(y)$ is correct. The same argument can be used to show that the proposition for $f_{X^i}(x)$ is also true.

Finally, we notice that the evaluation of Equations 5.12 and 5.13 corresponds to compounding the conditional distributions presented in Algorithm 4.

$$f_{Y^{i+1}}(y') = \sum_{x' \in R_X} f_{X^i}(x')f_{Y|X}(y'|x') , \quad \forall y' \in R_Y$$

$$f_{X^{i+1}}(x') = \sum_{y' \in R_Y} f_{Y^i}(y')f_{X|Y}(x'|y') , \quad \forall x' \in R_X$$

By using the degenerate Dirichlet distribution, we transform the target density distributions to be $f_{g(X)}$ and $f_{g(Y)}$, so that the method is guaranteed to converge using the following equations:

$$f_{g(Y^{i+1})}(y') = \int_{x' \in R_{g(X)}} f_{g(X^i)} F(x') f_{Y|g(X)}(y'|x') , \quad \forall y' \in R_Y$$

$$f_{g(X^{i+1})}(x') = \int_{y' \in R_{g(Y)}} f_{g(Y^i)}(y') f_{X|g(Y)}(x'|y') , \quad \forall x' \in R_X$$

Although the convergence may be biased to a different marginal distribution, we emphasize that this is perhaps beneficial in topic models such as LDA.

5.1.3 Relation with mean field variational inference

Variational inference approximates the posterior $P(Z_1, \dots, Z_m | \mathbf{X})$ using an optimization approach. The solution is given by fitting a proposed distribution $q(Z)$ to the posterior as measured by the Kullback-Leibler (KL) divergence. The relationship between $q(Z)$ and $P(Z | \mathbf{X})$ is given as follows:

$$\log p(\mathbf{X}) = ELBO(q) + D_{KL}(q_\gamma(Z) || p(Z | \mathbf{X})) \quad (5.14)$$

$$\text{where } ELBO(q) = \mathbb{E}_q [\log p(Z, \mathbf{X})] + \mathbb{E}_q [\log q(Z)]$$

Since the evidence $P(\mathbf{X})$ is strictly positive and independent of the latent variables Z , maximizing the $ELBO(q)$ is equivalent to minimizing the KL divergence. A common algorithm used to maximize the ELBO is coordinate ascent variational inference (CAVI) presented in Algorithm 5 [11].

The relationship between variational inference and our proposed method becomes visible if we propose the variational family $q_j(Z_j) \sim \text{deg-Dir}(\eta_j)$ to approximate the

Algorithm 5 Coordinate ascent variational inference (CAVI)

```

1: Input A model  $p(\mathbf{X}, Z)$ , a dataset  $\mathbf{X}$ 
2: Initialize Variational factors  $q_j(Z_j)$ 
3: while ELBO(q) has not converged do
4:   for  $j \in 1 \dots |\mathbf{X}|$  do
5:     Set  $q_j(Z_j) \propto \exp [\mathbb{E}_{q_{-j}} [\log p(Z_j | Z_{-j}, \mathbf{X})]]$ 
6:   end for
7:   Compute  $ELBO(q) = \mathbb{E}_q [\log p(Z, \mathbf{X})] + \mathbb{E}_q [\log q(Z)]$ 
8: end while
9: return  $q(Z) = \prod_{j=1}^M q_j(Z_j)$ 

```

posterior of RCA presented in Equation 5.4. Assuming the density of $q_j(Z_j)$ as $f_{Z_j}(z)$, step 5 of Algorithm 5 turns to be:

$$q_j(Z_j) \propto \exp \left(\int \cdots \int \log f_{Z_j | Z_{-j}, \mathbf{X}}(z) \prod_{m=1, m \neq j}^M f_{Z_j}(z) dz_{-j} \right) \quad (5.15)$$

The integrals of Equation 5.15 are different than zero only when $Z_j = \eta_j \forall j \in 1 \dots M$.

Therefore, we obtain that:

$$q_j(Z_j) = \exp \left(\log \left(f_{Z_j | Z_{-j}}(z | \eta_j) \right) \right)$$

This equation makes CAVI (Algorithm 5) become identical to RCA (Algorithm 6). In addition, because CAVI converges to the same objective of the Gibbs method, we notice that CAVI does a perfect approximation to the posterior of RCA when we initialize the variational family with degenerate-Dirichlet distributions. This observation may be used to transfer existing advantages of the mean-field variational inference algorithm into the Gibbs sampling framework.

Fig. 5.2 Approximation of the posterior $p(Z_j|Z_{-j} = z_{-j}, \mathbb{D})$ via sampling. The figure shows examples of m draws from the posterior. As the number of samples grows, we obtain better estimate of the distribution. The samples approach the posterior itself when m grows up to infinity.

5.2 Heron Inference

5.2.1 State augmentation methods

Gibbs sampling methods have been shown to achieve decent predictive accuracy. Nonetheless, Zhao and Canny proposed an state augmentation approach (SAME)[117] that draws several samples as opposed to drawing one single sample from each conditional distribution. This effectively corresponds to “cooling” the posterior, allowing an annealed search of the MAP parameters. As a result, the inference often yields higher-quality estimates than the single sample based approaches [30]. We illustrate this effect in Figure 5.2

Effectively, the idea of the replication of the latent state comprises taking m samples of the collapsed Gibbs sampler, and then computing the statistics of the model taking into account the n^{th} augmentation as follows:

$$\dot{C}_{k,d_j} = \frac{1}{m} \sum_{n=1}^m C_{k,d_j}^{(n)} \quad ; \quad \dot{D}_{k,w_j} = \frac{1}{m} \sum_{n=1}^m D_{k,w_j}^{(n)} \quad (5.16)$$

While, arguably the predictive power of this method is higher than CGS, the computational complexity has also been increased. To deal with the higher cost of obtaining additional samples, SAME employs a coordinated-factored approximation, in which it replaces sampling m multinomial distributions with a single Poisson sample per category, hence the label of coordinated approximation.

The efficiency of a Poisson sampler remains similar to sampling from a multinomial distribution as long as m takes a value of 100 or less. The sampling scheme replaces Line 7 and 8 in Algorithm 2, as shown below:

$$\begin{aligned} k'_j &\sim \text{Poisson}(m * p(Z_j = k | Z_{-j} = z_{-j}, \mathbb{D})) \\ \dot{C}_{k,d_j} &= \dot{C}_{k,d_j} + \frac{k'_j}{m} ; \quad \dot{D}_{k,w_j} = \dot{D}_{k,w_j} + \frac{k'_j}{m} \quad \forall k \in 1 \cdots K \end{aligned} \tag{5.17}$$

where k'_j (k^{th} coordinate topic assignment of the j^{th} record).

5.2.2 Maximization of the latent state replication

We improve SAME, by maximizing the number of state replications. Conveniently, as the number of replications approach to infinity, the Poisson distribution used by SAME converges to a unique value with probability of one. Therefore, in the end, sampling was not required and the Gibbs sampling algorithm can be transformed once again into a deterministic method. Interestingly, the obtained deterministic method is equivalent to the algorithm found by RCA in Section 5.1. By describing the connection of RCA with a maximal latent state replication method, we can theoretically justify the high predictive power of RCA. Next, we present the proof for this connection:

The two steps presented in Equation 5.17 can be reduced to one step by deriving the distribution from the linear transformation of the Poisson distribution when multiplied by $\frac{1}{m}$.

Let Y be a $Poisson(m\lambda_k)$ where $\lambda_k = p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D})$. Hence the probability mass of Y is equals to:

$$p(Y = y) = \frac{m\lambda_k^y e^{-m\lambda_k}}{y!} \quad (5.18)$$

Our objective is to find U as the linear transformation of Y such that $U = \frac{1}{m}Y$. So, we start by finding the relationship between the cumulative distributive function (CDF) of Y and U :

$$\begin{aligned} p(U \leq u) &= p\left(\frac{Y}{m} \leq u\right) \\ &= p(Y \leq mu) \\ &= F_y(mu) = F_u(u) \end{aligned} \quad (5.19)$$

Then, we calculate the probability mass function of U in terms of Y using the CDF's described above:

$$\begin{aligned} p(U = u) &= p(U \leq u) - p(U \leq u - 1) \\ &= F_u(u) - F_u(u - 1) \\ &= F_y(mu) - F_y(m(u - 1)) \\ &= \sum_{n=0}^{mu} \frac{m\lambda_k^n e^{-m\lambda_k}}{n!} - \sum_{n=0}^{m(u-1)} \frac{m\lambda_k^n e^{-m\lambda_k}}{n!} \\ &= \frac{m\lambda_k^{mu} e^{-m\lambda_k}}{mu!} \end{aligned} \quad (5.20)$$

To study the effect of increasing the number of latent replications, we compute the mean and variance of U when increasing m to infinity. Using the linearity of expectation we have that:

$$E[U] = \frac{E[Y]}{m} = \frac{m\lambda_k}{m} = p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D}) \quad (5.21)$$

So the mean of the distribution is independent of m . This is a desired result because the value of m should not add a shifting term to the topic assignment of the j^{th} document-word pair. As for the variance of U , we define the following expression:

$$Var[U] = \frac{Var[Y]}{m^2} = \frac{\lambda_k}{m} \quad (5.22)$$

The result shows that the variance tends to zero as m grows up to infinity. The conclusion is that when maximizing m , the probability mass of U starts to concentrate in the mean value λ_k . Figure 5.3 illustrates this effect.

Fig. 5.3 Behavior of $p(U)$ as $m \rightarrow \infty$ for $\lambda_k = 0.3$

Given the benefits of the latent state replication, the topic assignment of the j^{th} document-pair should be λ_k with probability 1. Therefore, by maximizing m , we obtain an update that does not require any sampling. Then, Line 7 and 8 of Algorithm 2 can

be replaced by:

$$\begin{aligned}\ddot{C}_{k,d_j} &= \ddot{C}_{k,d_j} + p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D}) \\ \ddot{D}_{k,w_j} &= \ddot{D}_{k,w_j} + p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D}) \forall k \in 1 \cdots K\end{aligned}\tag{5.23}$$

where

$$\ddot{C}_{k,d_j} = \sum_{n=1, n \neq j}^{|\mathbb{D}|} \mathbb{1}(d_j = d_n) p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D})\tag{5.24}$$

$$\ddot{D}_{k,w_j} = \sum_{n=1, n \neq j}^{|\mathbb{D}|} \mathbb{1}(w_j = w_n) p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D})\tag{5.25}$$

$$\ddot{N}_k = \sum_{n=1, n \neq j}^{|\mathbb{D}|} p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D})\tag{5.26}$$

To complete our proof, Algorithm 6 is equivalent to Algorithm 4 because both algorithms updates the same statistics. For the comparison, RCA statistics are presented by Equation 5.1.1, while the statistics of the maximal latent state replication algorithm are presented by Equations 5.24, 5.25, 5.26.

Algorithm 6 CGS with Maximum latent state replications

- 1: **Input:** Dataset $\mathbb{D}$ (size: $|\mathbb{D}|$), latent assignment k_j
 - 2: **Initialize** k_j , $\forall k_j \in 1 \cdots |\mathbb{D}|$, $\Omega = \text{computeStats}(k_j)$
 - 3: **repeat**
 - 4: **for** $j = 1$ **to** $|\mathbb{D}|$ **do**
 - 5: $\text{updateStats}(\Omega, \lambda_j, \text{Subtract})$
 - 6: $\lambda_j \propto g(\Omega)$ from Table 3.3
 - 7: $\text{updateStats}(\Omega, \lambda'_k, \text{Add})$ from Equation 5.23
 - 8: **end for**
 - 9: **until** convergence
-

Following RCA, the proposed Heron inference leverages multiple samples to “cool” the posterior. Differently, Heron analyzes the convergence properties of RCA presented

in Algorithm 6. When applying the convergence properties to RCA, we obtain a system of equations that may be solved via fixed-point iterations.

5.2.3 Convergence properties of Gibbs sampling

Typically, Gibbs sampling is used to estimate marginal distributions from joint distributions. For instance, given that we know $p(X, Y)$, it may be analytically intractable to find the marginal distribution $p(X)$ or $p(Y)$. In this case, we resort to methods such as Gibbs sampling to estimate the marginals from conditionals as follow:

1. Find an initial sample x_0
2. Sample y_1 from $p(Y|X = x_0)$
3. Sample x_1 from $p(X|Y = y_1)$
4. Reiterate

As shown by Casella and George [19], this process converges to find samples from the marginal distributions $p(X)$, and $p(Y)$. Assuming discrete distributions, the basic idea is that one iteration of Gibbs sampling (Steps 2 and 3 from the above procedure) transforms an initial distribution $p(X_0)$ into $p(X_1)$ using a transition matrix $A_{X|X} = p(X|Y)p(Y|X)$ as follows:

(5.27)

After a number of iterations, the theory of Markov chains states that the marginal distribution $p(X)$ will stop transforming, at which point, Gibbs sampling flags the

convergence of the algorithm and Equation 5.2.3 simply becomes:

$$p(X^*) = p(X^*)A(X|X) \quad (5.28)$$

$p(X^*)$ is known as the stationary distribution of the Markov chain $A_{X|X}$. Going back to Algorithm 6, we can enumerate the steps of the algorithm as follows:

Initialize $\lambda_{jk}^0 \forall j \in 1..|\mathbb{D}|, \forall k \in 1..K$

—— Iteration 0

$$\begin{aligned} \lambda_{1k}^1 &= g(\lambda_{-1k}^0) & \forall k \in 1..K \\ \lambda_{2k}^1 &= g(\lambda_{1k}^1, \lambda_{>2k}^0) & \forall k \in 1..K \\ & \vdots \\ \lambda_{|\mathbb{D}|k}^1 &= g(\lambda_{-|\mathbb{D}|k}^1) & \forall k \in 1..K \end{aligned} \quad (5.29)$$

—— Iteration 1

$$\begin{aligned} \lambda_{1k}^2 &= g(\lambda_{-1k}^1) & \forall k \in 1..K \\ \lambda_{2k}^2 &= g(\lambda_{1k}^2, \lambda_{>2k}^1) & \forall k \in 1..K \\ & \vdots \end{aligned} \quad (5.30)$$

—— At convergence, we have that

$$\begin{aligned} \lambda_{1k}^* &= g(\lambda_{-1k}^*) & \forall k \in 1..K \\ \lambda_{2k}^* &= g(\lambda_{-2k}^*) & \forall k \in 1..K \\ & \vdots \\ \lambda_{|\mathbb{D}|k}^* &= g(\lambda_{-|\mathbb{D}|k}^*) & \forall k \in 1..K \end{aligned} \quad (5.31)$$

where $g(\Omega)$ is the posterior equation derived for the model of interest (see Table 3.3). Note that Ω is computed based on the topic assignments for each coordinate as presented in Equations 5.24, 5.25, and 5.26.

Rather than computing the iterative steps of CGS with maximum state replication, we instead aim at directly solving the system of equations that must hold at the convergence of the algorithm. With a simple mathematical manipulation, we can show that CGS with maximal state replication is ultimately seeking a fixed-point equation of the form $x = f(x)$. We define x_{dwk} , as the unknown parameter of the equation representing $p(Z_j = k | Z_{-j} = k_{-j}, \mathbb{D})$ where j indexes the document-word tuple composed by document d and word w . Applying the system of equations 5.31 to LDA, we obtain:

$$x_{dwk} = \frac{\frac{(\ddot{C}_{dk} + \alpha)(\ddot{D}_{wk} + \beta)}{\ddot{N}_k}}{\sum_{k'=1}^{|K|} \frac{(\ddot{C}_{dk'} + \alpha)(\ddot{D}_{wk'} + \beta)}{\ddot{N}_{k'}}} \quad (5.32)$$

where

$$\ddot{C}_{dk} = \sum_j^{|D|} x_{dwk} \quad ; \quad \ddot{D}_{wk} = \sum_i^{|C|} x_{dwk} \quad ; \quad \ddot{N}_k = \sum_j^{|D|} \sum_i^{|C|} x_{dwk}$$

One advantage of the system of equations is that the order used to enumerate the equations is irrelevant because any ordering yields the same solution. This is not the case for the sampler counterparts, where the order of sampling affects the convergence results [51]. In addition, the sequential nature of Gibbs sampling complicates the implementation of distributable algorithms.

We have transformed the inference problem into a system of $|D| \times |Nd| \times K$ non-linear equations whose solutions correspond to all possible outcomes of an infinitely replicated latent state augmentation.

5.2.4 Resolving the sampling of repeated tuples

Finally, we can further simplify the system of equations by observing that the same document-word tuples will also converge to the same values. Thus, we have:

$$P(Z_j|Z_{-j}, X) = P(Z_m|Z_{-m}, \mathbb{D}) \quad \forall j \in 1..|\mathbb{D}| \quad (5.33)$$

such that $d_j = d_m \wedge w_j = w_m$

We can validate Equation 5.33 by using Equation 5.32. Upon convergence, repeated tuples will have the same fixed-point equations. Therefore, we remove the repeated equations and the problem is transformed into solving a system of $|D| \times |W| \times K$ non-linear equations. Also, given that the unknown values x_{dwk} are the same for all repeated document-word pairs, we may compact the statistics of Equation 5.32 as follows:

$$\ddot{C}_{dk} = \sum_j^{|D|} x_{dwk} c_{dw}; \ddot{D}_{wk} = \sum_i^{|C|} x_{dwk} c_{dw}; \ddot{N}_k = \sum_j^{|D|} \sum_i^{|C|} x_{dwk} c_{dw} \quad (5.34)$$

where c_{ij} correspond to the count of how many times the i^{th} word of the vocabulary has appeared in the j^{th} document.

We use the fixed-point iterative method to solve the system of equations. Applying the mean value theorem of calculus, the fixed-point method finds the solution to a system of equations $x = f(x)$ as long as f and f' are continuous functions.

The algorithm is similar to CGS presented in Algorithm 2 with the main differences: 1) There is no need to iterate over $|D| \times |Nd|$ steps, and 2) The algorithm is distributable since we only need to update the statistics until the end of each iteration. Theoretically, we can partition the dataset into as many pieces as $|D| \times |W|$. Independently of how we partition the dataset, the estimated parameters will be the same as long as we combine the results at the end of each iteration. We present Heron inference in Algorithm 7. Note that the definition of $\Omega = \{C, D, N\}$ is overridden with $\ddot{\Omega} = \{\ddot{C}, \ddot{D}, \ddot{N}\}$.

Algorithm 7 Heron Inference

```

1: Input: latent assignment  $z_j$ , dataset size  $m$ ,
2: Initialize  $x_{dwk} \forall d \in 1..|C| \forall j \in ..|D| \forall k \in 1..K$ 
3: Compute  $\ddot{\Omega}$  from Equation 5.34
4: repeat
5:   for  $d = 1$  to  $|D|$  do in parallel
6:     for  $w = 1$  to  $|W|$  do in parallel
7:       for  $k = 1$  to  $K$  do in parallel
8:          $x_{dwk} \propto g(\ddot{\Omega})$  from Table 3.3
9:       end for
10:    end for
11:  end for
12:  Compute  $\ddot{\Omega}$  from Equation 5.34
13: until convergence (Equation 5.35)

```

An additional advantage of the deterministic Heron method is that the assessment for convergence is now possible within the Gibbs framework. We are allowed to explore the assessment for convergence using the average Kullback-Leibler divergence (KL) or the average Chi-Squared distance on the learned θ and ϕ . We find that the latter metric oscillates and it does not have a monotonic behavior. We find that it is often more practical to measure the change in either θ or ϕ . Equation 5.35 presents the metric used to assess convergence of Heron at the i^{th} iteration.

$$D_{KL}(\theta^i || \theta^{i-1}) = \frac{1}{|C|} \sum_{d=1}^{|C|} \theta_d^i \log\left(\frac{\theta_d^i}{\theta_d^{i-1}}\right) \quad (5.35)$$

Finally, the state augmentation based methods are under the premise of independence of the latent variables given the other parameters of the model, therefore the methods such as hidden Markov model [7] and stochastic block mixture model [1] are not guaranteed to work using these inference approaches.

* Equivalent to variational inference with a particular choice of variational family

Fig. 5.4 Connection between the inferential methods.

5.3 Contribution to the Inference of Topic Models

To present our contribution, we use Figure 5.4 to review how we achieved a better inference model in terms of computational efficiency and predictive power.

We have started our study with the collapsed Gibbs sampling algorithm (CGS) for the learning topic models which iterates following a point estimate. Then, we show that an alternative to CGS is the substitution algorithm which iterates using the entire distribution and eases the assessment of convergence. However, we found that this alternative is often intractable when applied to some probabilistic models such as LDA. To cope with this issue, we introduced a function that transforms the categorical distribution into a degenerate Dirichlet distribution. This transformation yields a tractable solution for the substitution algorithm that provides topic models with an additional flexibility in topic assignments. The experimental results verify that RCA achieves overall improvement over CGS in terms of both topic coherence and perplexity despite the setting of the hyperparameters.

From a different branch, we follow an inference method that learns by using multiple point estimates called state augmentation for marginal estimation (SAME). This method is able to improve the predictive power of CGS. In addition, SAME uses a Poisson sampler to maintain the computational complexity of CGS and to allow a parallel implementation of the model which results in an enhanced computational efficiency. However, SAME can only approximate a limited number of multiple point estimates. In our study, we find mathematically that by increasing the point estimates up to infinity, the algorithm becomes deterministic and the equations yield the same equations of RCA. Intuitively, iterations driven by a very large number of samples are using the entire information given by the conditional distribution, and this is the same idea of RCA's soft-topic assignments provided via the degenerate Dirichlet transformation. Although the two branches converge into the same method. We find a benefit from studying the two branches. While the Dirichlet transformation leverages on the substitution algorithm to guarantee the convergence of the algorithm and explain how Gibbs sampling may turn into a deterministic algorithm, the state augmentation branch presents a theoretical argument on the improvement of the predictive power of RCA.

Unfortunately RCA, just like CGS, is non trivially distributable because every step in the algorithm depends on the results from the previous step. We solved this issue by studying the properties of convergence of RCA. The determinism of the method facilitates this process and we discover that CGS is ultimately describing a system of non-linear equations. The system of linear equations can be used to avoid the computation of duplicated tuples and more importantly, We employ a fixed-point iterative method to solve the system of equations. The method is distributable and uses GPU resources more efficiently achieving a novel inference method with higher computational efficiency than SAME [117].

Finally, we note that the axes of the figure do not represent a numerical scale but they only illustrate the ordering in computational efficiency and predictive power of the inference methods. A thorough evaluation is presented in Section 5.4.

5.4 Experimental Evaluation

5.4.1 Datasets

We used five popular and publicly available datasets for our experimental evaluation.

- **Cora** dataset contains research papers and their citation networks. After preprocessing, we obtained 2,618 documents with a vocabulary composed of 1,400 words. The number of citation links is 5,212. The dataset was first used for testing the RTM model [22] ².
- **Diggs** dataset consists of 6,192 document-rating pairs with a vocabulary composed of 3,145 words after preprocessing. Note that, following [73], we normalized the ratings into the range $[1, 5]$ to avoid numerical overflows when calculating sLDA's posterior. The dataset was originally used for evaluating sLDA [73] ³.
- **MovieLens** is a popular benchmark for rating prediction and recommendation. We used the MovieLens 20M dataset ⁴, and focused on the most popular 100,000 users who rated 25,646 movies. We treated the movies and users as pseudo documents and words, respectively. The average rating of all the related users on a movie was used as the rating of the movie (document).

²<https://people.cs.umass.edu/~mccallum/data.html>

³<http://www.cs.columbia.edu/~blei/papers/digg-data.tgz>

⁴<https://grouplens.org/datasets/movielens/>

- **NIPS** dataset comes from the Neural Information Processing Systems proceedings from 1987 to 2015. The dataset consists of 11,040,357 records, which are generated from 5,811 authors and 11,463 words ⁵.
- **Lastfm** dataset contains 145,534,518 records which are generated from the tuples (user, artist, plays) for 360,000 users ⁶.

We evaluate our proposed maximal latent state replication inference method (Heron) against two well-established baseline methods. We do not compare RCA method because the convergence of RCA is the same as the convergence of Heron. In effect, RCA can be thought of as a stepping stone in the development of Heron. In this subsection we aim to answer the following research questions (**RQs**)

RQ1 Is Heron able to fit better a test set than the state-of-the-art inference method SAME?

RQ1a Is Heron less sensitive to the hyper parameter choice than the other inference algorithms?

RQ2 What is the effect on the predictive perplexity as we increase the number of batches, and does the moving average estimate of θ improve Heron’s learning?

RQ3 How sensitive is Heron to the setting of the hyperparameters α and β ?

RQ4 How does the efficiency of Heron compare against the arguably fastest GPU-based inference method within the Gibbs framework?

RQ5 How does the efficiency of Heron inference get affected by the batch size?

RQ6 Does Heron help extract topics with acceptable coherence?

⁵<https://archive.ics.uci.edu/ml/datasets/NIPS+Conference+Papers+1987-2015>

⁶<https://grouplens.org/datasets/movielens/>

RQ7 Can Heron improve the inference of DT2B model?

Baselines We compared Heron inference with two strong baselines, i.e., standard collapsed Gibbs sampling and SAME inference method [117]. collapsed Gibbs sampling is perhaps one of the most commonly used methods for inferring graphical models, while SAME has been shown to attain the best predictive perplexity and computational efficiency for inferring LDA model.

Topic Models To test the performance of Heron for inferring graphical models, we used three well-known topic models, i.e., LDA (latent Dirichlet allocation) [13], RTM (relational topic model) [22] and sLDA (supervised latent Dirichlet allocation) [73]. For LDA, all the datasets can be used to evaluate the inference methods. Inference for RTM can be only tested on the Cora dataset, as only Cora has a citation networks. We used Movielens and Diggs to test the inference for sLDA, since in addition to documents/movies, corresponding ratings are also available.

Implementation Details All the algorithms were evaluated on a single PC equipped with a single 2-core CPU (Intel Xeon @ 2.20 GHz) and a *Nvidia tesla K80* GPU with dual stream processors. Each GPU processor comes with 2,496 cores and 12GB of GDDR5 clocked at 5GHz. Only one processor of the GPU was used in the benchmark. The standard Gibbs sampler was implemented using CPU resources. In all our experiments we fixed both the grid size and block size.

We assess our proposed inference method in terms of running time, predictability, as well as coherence of the extracted topics. In particular, we conduct the following three experiments.

5.4.2 Predictive perplexity of topic models

The problem defined is given as follows: given a fraction of the original data, we evaluate our model’s ability to generate the withheld portion. As such, the evaluation

Table 5.1 Hyperparameter setting per dataset.

Dataset	α	β	η	a
Cora	0.4	0.3	0.75	N/A
Diggs	0.6	0.5	0.5	0.25
Movielens	0.6	0.6	0.5	0.4

Fig. 5.5 Predictive perplexity on the GPU based inference methods versus the number of document-word pairs per batch.

metric adopted in our experiments is the predictive perplexity. The number of topics K is tuned amongst $\{10, 25, 50, 100\}$. The training-testing split is 70 – 30, and we run all models for 1,000 iterations. Furthermore, we conducted five trials for each setting in the spirit of robust experimentation, and we reported the average results. The hyperparameters are given in Table 5.1. Models applied to the same dataset use the same hyperparameter settings.

Table 5.2 shows the comparison results of all the inference methods when applied to the three models. Answering **RQ1**, our proposed Heron inference outperforms CGS and SAME method, i.e., both the best and second best performance on each configuration comes from Heron model across all K values. As expected, SAME method consistently outperforms CGS. This is due to the fact that SAME has been derived from CGS by applying a finite replication of the latent states, and Heron maximizes this effect using

Table 5.2 Predictive perplexity versus the number of topics.

	Model	Inference	Perplexity			
			K=10	K=25	K=50	K=100
Cora	LDA	CGS	1709.63	1837.46	1999.25	2623.86
		SAME	1706.27	1821.20	1977.26	2549.43
		Heron	1683.03	1770.66	1977.51	2550.21
	RTM	CGS	1700.12	1830.02	1995.37	2245.96
		SAME	1699.20	1811.96	1991.12	2043.35
		Heron	1682.32	1806.09	1991.02	1971.07
Diggs	LDA	CGS	952.18	1166.93	1500.51	2075.65
		SAME	948.57	1161.40	1492.53	2067.24
		Heron	947.27	1160.35	1489.43	2062.12
	sLDA	CGS	940.09	1125.85	1452.94	2007.00
		SAME	919.31	1123.82	1446.41	2005.28
		Heron	914.77	1121.96	1442.54	2002.61
Movielens	LDA	CGS	1085.47	1022.25	1011.62	1065.02
		SAME	924.12	932.43	993.81	1045.75
		Heron	922.20	931.34	989.92	1045.09
	sLDA	CGS	926.55	905.56	924.75	963.45
		SAME	923.28	903.63	921.26	963.65
		Heron	920.22	899.68	918.51	954.28

Fig. 5.6 Predictive perplexity on the GPU based inference methods versus the number of document-word pairs per batch.

a mathematical manipulation. SAME method improves Gibbs sampling because the estimation for the parameters is done based on the document-topic and word-topic counts. Hence, the estimation is based on discrete values. By using multiple replications of the latent state, it is possible to make this estimation more granular so that the counts can be thought to reside in the space of real numbers. As we increase the number of replications, the estimation becomes more granular, and Heron becomes the most flexible approach approximating the local optima closer than the other methods.

The benefit of the distributed methods lies in their ability to split the dataset into batches, so that several GPUs can be used to boost the speed of inference. As such, we propose **RQ2** and solve it by studying the effect of splitting the datasets into different batch sizes. We fix $K = 25$ and split the dataset into exponentially larger number of batches. Similar to SAME method, Heron inference may treat the batches as an infinite stream, and it is not necessary to denote passes over the dataset. In addition, we also study SAME’s moving average estimate for θ [117], which has shown improvements when applied to other learning algorithms [52]. We label the results of applying this technique as Heron-LDA*, Heron-sLDA*, and Heron-RTM*.

We answer **RQ2** using Figures 5.5 and 5.6. We conclude that the batched versions converge to a better estimation as measured by the predictive perplexity. As the dataset is split into smaller pieces, the predictive perplexity improves. This may be due to the fact that splitting the dataset into batches allows the inference to reach a better local optima with the use of a moving average estimate for θ [117, 52]. Nonetheless, after a certain number of batches the predictive perplexity is affected, indicating that the learning has become compromised. Note that Heron can be distributed into as many batches as the number of document-word tuples. The final inference will result in the same estimation as if we partitioned the dataset into any number of pieces or if we did not partition the dataset. This is because Heron inference is ultimately solving

a fixed-point system of equations, given by Equation 5.32. The results show that Heron can yield better local optima if we decide to use the moving average updates. In that case, the learning of Heron may also be compromised if we split the dataset into a large number of batches.

5.4.3 Hyperparameter sensitivity

Table 5.3 Predictive perplexity versus the hyperparameters α and β .

LDA - CORA		$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.3$	$\beta = 0.4$	$\beta = 0.5$
CGS	$\alpha = 0.1$	2118.8	1974.42	1766.27	1741.98	1667.46
	$\alpha = 0.2$	1954.09	1831.44	1751.84	1725.55	1677.72
	$\alpha = 0.3$	1880.51	1806.89	1747.28	1735.68	1748.78
	$\alpha = 0.4$	1912.38	1851.29	1784.29	1784.60	1784.55
	$\alpha = 0.5$	1916.95	1882.72	1845.47	1834.62	1845.15
SAME	$\alpha = 0.1$	2102.48	1916.14	1721.83	1649.48	1656.24
	$\alpha = 0.2$	1942.4	1775.68	1713.5	1664.11	1656.55
	$\alpha = 0.3$	1824.89	1794.71	1735.07	1707.69	1719.56
	$\alpha = 0.4$	1840.68	1779.86	1796.25	1787.89	1798.68
	$\alpha = 0.5$	1898.44	1850.68	1845.86	1860.4	1866.48
Heron	$\alpha = 0.1$	1858.19	1738.36	1666.94	1647.12	1635.79
	$\alpha = 0.2$	1751.62	1704.31	1656.84	1661.41	1656.67
	$\alpha = 0.3$	1708.78	1655.16	1671.78	1692.64	1702.59
	$\alpha = 0.4$	1745.76	1739.17	1737.66	1750.46	1775.52
	$\alpha = 0.5$	1801.18	1767.2	1782.23	1831.43	1847.63

As proposed in **RQ3**, we studied the hyperparameter sensitivity of the different inference methods. We made a grid search on α and β starting at 0.1 and stopping at 0.5 with steps of 0.1. In addition, we fixed $K = 25$ and we let the methods run for 500 iterations to guarantee a fair assessment. For the evaluation, we applied the inference methods to LDA on the Cora dataset.

Table 5.3 shows that the predictive perplexity of Heron is superior to that of SAME, and as expected, SAME has more predictive power than CGS. Given the same

combination of α and β , Heron significantly improves SAME and SAME consistently improves over CGS.

Moreover, we observe that the inference of LDA via CGS and Heron is more sensitive to the choice of values of the hyperparameters. The gap between the maximum and minimum perplexity scores is 451.34 for CGS, 453.0 for SAME, and the gap is only 222.4 for Heron. This also matches our expectation, as CGS is known to be sensitive to the hyperparameter setting. Based on the range of the perplexity across different hyperparameters, we conclude that Heron is more robust to the hyperparameter settings, while CGS and SAME have similar sensitivity.

5.4.4 Learning time

In this subsection, we report the runtime to convergence of the inference algorithms normalized by the number of iterations.

First, we study the effect of the running time given changes in the number of topics. As shown in Table 5.4, Heron is about one order of magnitude faster than the state-of-the-art SAME method across all the values of K .

The main reason lies in Heron does not rely on the sampling operations for inference. In contrast, SAME method not only incurs the sampling costs but also is limited at every clock cycle by the available number of samplers within the GPU. Benefiting from purely arithmetic operations, Heron does not suffer from the limitations and sampling costs. We also observe that the running time increases linearly as we increase K for the three inference methods. While this increment is larger for Heron, Heron can also benefit from tuning the *grid size* and *block size* (GPU Kernel parameters) to favor different values of K . The flexibility of tuning GPU parameters is reduced for SAME, since at its optimal configuration, the system will be still limited by the availability

Table 5.4 Average running time per iteration versus the number of topics.

	Model	Inference	Running time (μs)			
			K=10	K=25	K=50	K=100
Cora	LDA	RCA	1375	1412	1481	1749
		CGS	1350	1390	1460	1728
		SAME	190	214	231	291
		Heron	3	7	11	18
	RTM	RCA	1564	1643	1711	1740
		CGS	1540	1620	1690	1720
		SAME	210	218	222	240
		Heron	5	12	19	33
Diggs	LDA	RCA	2619	3167	3055	3324
		CGS	2590	3140	3030	3300
		SAME	260	290	270	330
		Heron	13	15	18	31
	sLDA	RCA	4089	4116	4494	5142
		CGS	4060	4090	4470	5120
		SAME	270	280	280	320
		Heron	15	23	30	50
Movielens	LDA	RCA	16483	17431	18602	21356
		CGS	16405	17356	18511	21286
		SAME	780	790	813	833
		Heron	19	44	62	95
	sLDA	RCA	7096	7863	9000	10401
		CGS	7053	7818	8954	10367
		SAME	954	976	997	1028
		Heron	23	46	67	102

of hardware based samplers. Finally, RCA and CGS have the same computational complexity and their running times are nearly the same. This answers **RQ4**.

In Figures 5.7 and 5.7, we show the effect of splitting the datasets into batches. The results show that both inference methods increase its running time logarithmically to the number of batches. This is outstanding given that Heron’s convergence is invariant to the number of batches, when the moving average estimate is not applied. It means that Heron can converge to accurate inference results dramatically fast, given that we

Fig. 5.7 Average running time per iteration versus the number of batches.

have enough cores running simultaneously. Answering **RQ5**, Heron efficiency reduces logarithmically to the number of batches slightly.

5.4.5 Automatic topic coherence

Besides assessing how the methods fit the data, it is important to evaluate the quality of the topics inferred by the methods. Following the evaluation introduced by Newman et al. [77], we computed the automatic topic coherence based on the top N most likely words for each of the extracted topics. Specifically, we relied on word co-occurrence metrics on the corpus to compute the average coherence of each topic. The automatic topic coherence has been studied extensively, and the normalized point-wise mutual information (NPMI), the probabilistic mutual information (PMI), and the pairwise log-conditional probability (LCP) are co-occurrence metrics that have been shown to correlate positively to the human coherence evaluation [64]. Thus, we use these metrics to assess the coherence of the extracted topics.

For the experimental settings, we fixed α and β to 0.5, and the number of topics K to 25. We varied the value of N (10, 20, 50) to study the effect of defining the coherence

Fig. 5.8 Average running time per iteration versus the number of batches.

with different sets of topics. Finally, we normalized the measurements to be invariant to changes in N , so that it facilitates the comparison of the results. The following equation illustrates the normalization of the LCP metric for the k^{th} topic.

$$LCP(k) = \frac{2}{(N-1)*(N-2)} \sum_{j=2}^N \sum_{i=1}^{j-1} \log \left(\frac{P(w_j, w_i | k) + 1}{P(w_i | k)} \right)$$

Results presented in Table 5.5 shows the topic coherence results in LCP, PMI, and NPMI (The higher, the better). Overall, Heron is able to infer latent topics with much better or comparable coherence compared to that by CGS or SAME. Interestingly, SAME achieves the best topic coherence in Movielens for $N = 20$. Although, simultaneously, it also obtains the worst results for $N = 50$. Apparently, SAME assigns a high ranking to the most frequent words in the dataset which leads to an improved coherence as measured by the co-occurrence based metrics.

In all datasets, a smaller value of N results in better coherence. This can be justified by the fact that as we increase the number of words inside a topic, the harder it is to maintain the coherence of the topic. In other words, the top ranked words per topics typically result in better representation for the topic.

Table 5.5 Automatic topic coherence of LDA inferred by CGS, SAME, and Heron. Best results per dataset use bold font. From top to bottom, the datasets are Cora, Diggs, and Movielens, respectively.

	Method	N	LCP		NPMI		PMI	
			mean	std	mean	std	mean	std
Cora	CGS	20	-6.496	0.031	0.043	0.003	0.641	0.120
	SAME		-6.083	0.024	0.058	0.006	0.793	0.056
	Heron		-6.032	0.033	0.055	0.002	0.823	0.070
	CGS	50	-6.819	0.029	0.030	0.003	0.477	0.051
	SAME		-6.678	0.039	0.037	0.001	0.503	0.019
	Heron		-6.619	0.021	0.030	0.002	0.548	0.021
Diggs	CGS	20	-8.693	0.001	0.046	0.001	0.941	0.002
	SAME		-8.524	0.066	0.051	0.006	1.062	0.121
	Heron		-7.483	0.040	0.159	0.004	3.047	0.069
	CGS	50	-8.593	0.001	0.052	0.001	1.025	0.001
	SAME		-8.176	0.024	0.085	0.002	1.681	0.004
	Heron		-8.165	0.014	0.109	0.002	2.118	0.042
Movielens	CGS	20	-6.889	0.121	0.247	0.007	4.865	0.136
	SAME		-6.527	0.000	0.295	0.000	4.905	0.000
	Heron		-6.797	0.107	0.249	0.009	5.838	0.178
	CGS	50	-6.853	0.033	0.247	0.003	4.855	0.064
	SAME		-7.157	0.000	0.231	0.000	4.543	0.000
	Heron		-6.812	0.029	0.250	0.002	4.921	0.043

Given that Heron is able to improve CGS in all datasets, we conclude that Heron is able to infer coherent latent topics, answering **RQ6**.

5.4.6 Applying Heron inference to DT2B

Next, we evaluate Heron inference against CGS for the inference of the dual topic model (DT2B) using the predictive perplexity on Movielens, NIPS and LastFM datasets. We do not compare the results with SAME, the reason is that SAME requires the latent topics to be independent given the other parameters of the model and DT2B models the dependency between two different types of latent variables. Therefore, it is not clear how to implement SAME for this DT2B.

Fig. 5.9 Predictive perplexity of DT2B inferred by Heron or CGS for various values of Kr and Kc on the Lastfm dataset.

Fig. 5.10 Predictive perplexity of DT2B inferred by Heron or CGS for various values of Kr and Kc on the NIPS dataset.

Fig. 5.11 Predictive perplexity of DT2B inferred by Heron or CGS for various values of Kr and Kc on the Movielens dataset.

In DT2B, to estimate the probability of a document-word pair $p(w, D)$, we use the estimated parameters θ , ϕ_r , ϕ_c as follows:

$$p(w, D) = \sum_{kr=1}^{Kr} \sum_{kc=1}^{Kc} p(w|\phi_{kr})p(D|\phi_{kc})p(kr, kc|\theta)$$

In this experiment, we run a basic grid search ranging over the Kr and Kc values of 10 and 15 and we set the hyperparameters α , β_r , and β_c as 0.5. Figure 5.9, 5.10, and 5.11 present the predictive perplexity of DT2B as a function of the number of iterations for various values of Kr and Kc . Overall, Heron again outperforms CGS for inference of the topic model DT2B for various combinations of the Kr and Kc values on the three datasets. This serves to answer **RQ7**

It is worth noting that both inference methods take a few iterations before they begin to learn. This is because DT2B has been trying to simultaneously assign the values of two latent random variables. We also observe that the amount of iterations required for the methods to start learning depends on the initialization. A uniform initialization typically translates into more iterations taken by the methods to start learning. A good initialization may be to avoid an uniform initialization for θ , in this way, there will be a preference on the distribution assignments of the latent variables.

5.4.7 Document classification

The improvement in predictive perplexity attained by Heron Inference suggests an improved predictive power of topic models. With the intention of analyzing how does the predictive power of Heron translate to real applications, we have decided to evaluate our proposed method using document classification. Document classification consists in assigning one or more categories to every document in a collection so that the collection is easier to manage and sort. Given the importance of managing collections, document classification has been widely studied [84],[118]. In addition, the task has been modified

to fit other applications such as classification of short documents [109] and cross lingual document classification in which the objective is to learn a classification model from the labeled documents in one language and then apply it to the classification of documents in a second language [27].

Baselines As in the previous experiments, we compared Heron inference with two strong baselines, i.e., standard collapsed Gibbs sampling and SAME inference method [117]. For the classification task, we chose the discrete supervised LDA as this is an appropriate topic model for a supervised learning task.

Datasets We used two popular and publicly available datasets for our experimental evaluation.

- **N20** 20 Newsgroups dataset set consists of approximately 20000 newsgroup articles collected nearly evenly across 20 different newsgroups [62]. After removing stop words, selecting the top 5000 most frequently used words and filtering all attributes except ‘from’ and ‘subject’, we obtained 18828 documents with a vocabulary of size 5000. We split the documents so that 60% is used for training, 10% is used for validation and 30% are used for the test set. ⁷
- **R50** The Reuters dataset is a collection of 21578 documents that appeared on the Reuters newswire in 1987 and were manually classified by personnel from Reuters Ltd. For the preprocessing, we removed stopped words and we eliminated those documents without class label or with multiple labels. Different from N20 dataset where the documents are nearly balanced, the classes of Reuters are highly unbalanced. Hence we use 60% of the documents per class for training, 10% for validation and the remaining documents are used for the test set. After

⁷<http://archive.ics.uci.edu/ml/datasets/twenty+newsgroups>

Table 5.6 Hyperparameter values of sLDA as found for the classification of 20 news dataset using CGS, SAME, and Heron.

Method	K=25			K=50			K=75			K=100		
	α	β	a	α	β	a	α	β	a	α	β	a
CGS	1.4	0.45	2.5	1.25	0.35	2.0	0.8	0.3	2.0	0.4	0.2	2.0
SAME	1.3	0.4	2.0	1.25	0.3	2.0	0.8	0.25	1.5	0.3	0.2	1.5
Heron	1.3	0.4	2.0	1.2	0.3	2.0	0.8	0.25	1.5	0.25	0.15	1.5

Table 5.7 Hyperparameter values of sLDA as found for the classification of news articles using CGS, SAME, and Heron.

Method	K=25			K=50			K=75			K=100		
	α	β	a	α	β	a	α	β	a	α	β	a
CGS	1.5	0.8	3.5	1.4	0.7	3.5	1.1	0.9	4.0	1.2	0.9	3.5
SAME	1.2	0.7	2.5	1.2	0.75	2.5	1.0	0.7	2.5	1.0	0.6	2.5
Heron	0.8	0.7	2.0	0.9	0.75	2.5	1.0	0.65	2.0	0.9	0.7	2.0

eliminating those classes that could not be split into the three sets, we ended up with 50 classes, hence the label R50. ⁸

Hyperparameters

Because we are comparing the inference methods for the same probabilistic model, we do not set the same hyperparameter values α , β and a across the baselines. Specific hyperparameter settings may favour the learning of one of the baselines relative to the other baselines. Therefore, we use a validation set to tune the hyperparameters of sLDA for each of the baselines and we compare the model that yields the best accuracy for each of the inference methods. We use a grid search to find the best hyperparameters. We do not tune the number of topics, but instead we study the effect changing the number of topics on the application. For both datasets, we set K to be 25, 50, 75 and 100. The hyperparameters applied to the N20 and R50 datasets are presented in Tables 5.6 and 5.7 respectively.

⁸<http://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection>

Table 5.8 Micro-F1 score for the classification of newsgroups posts using sLDA as inferred by CGS, SAME, and Heron

Method	K=25		K=50		K=75		K=100	
	mean	std	mean	std	mean	std	mean	std
CGS	78.88	0.131	84.34	0.183	86.90	0.201	84.16	0.120
SAME	81.13	0.081	85.12	0.085	88.88	0.072	88.62	0.090
Heron	84.11	0.041	89.83	0.033	93.13	0.038	92.21	0.042
Improvement	2.98%		4.71%		4.25%		3.59%	

Table 5.9 Micro-F1 score for the classification of Reuters news articles using sLDA as inferred by CGS, SAME, and Heron

Method	K=25		K=50		K=75		K=100	
	mean	std	mean	std	mean	std	mean	std
CGS	47.91	0.069	48.24	0.055	48.12	0.078	47.58	0.071
SAME	48.52	0.061	48.55	0.065	48.56	0.059	48.29	0.051
Heron	50.68	0.041	51.12	0.043	50.81	0.045	50.17	0.038
Improvement	2.16%		2.57%		2.25%		1.88%	

Evaluation metrics

With the spirit of robust experimentation, we run the inference for the N20 and R50 datasets ten times, every time under a different initialization. For both datasets, we report the average Micro-F1 score and its standard deviation in Tables 5.8 and 5.9. The Micro-F1 score is selected instead as opposed of the more popular Macro-F1 score given that the Micro F1 accounts for the class unbalancing [72]. This is particularly important in the R50 dataset, where the two most popular classes account for around 70% of the entire dataset.

Results

Tables 5.8 and 5.9 show an overall improvement of Heron over SAME which in turn improves over CGS. This is consistent with the findings by Canny et al. [117]. The predictive power of SAME and Heron increases because of the latent state augmentation which leads the inference algorithms to converge towards the set of maximum a posteriori

parameters [2]. This result holds across the set of studied K values. The standard deviation values trivially show that the improvement is statistically significant. In fact, we observed that the best run of CGS was not able to improve the worse run of Heron inference. We also noticed that the improvement in the N20 dataset is overall greater than the improvement in the R50 dataset. Our hypothesis is that Heron is able to fine tune the values of the parameters obtained by SAME and CGS so that it consistently improves the predictive power of the other inference algorithms. However, the fine tuning achieved by Heron may not be enough to drastically change the values of the parameters and yield dramatic improvements over other algorithms in the Gibbs family. Our hypothesis is reinforced by observing at the improvement obtained for a low number of topics. For $K = 25$, The improvement is lower in both datasets. A low value of K implies that less features are used during the classification. Therefore, the predictive power may be affected more by the number of features than by the tuning of the trained parameters.

As presented by Blei et al., the function of the predictive power of topic models with respect to the number of topics has a concave behaviour [60]. Therefore, we expect a value of K to produce the optimal results for each algorithm. For N20 dataset, all the inference methods show a preference for $K = 75$, and $K = 50$ for the R50 dataset. Apart from K , the remaining hyperparameters shown in Tables 5.6 and 5.7 have a larger influence in the tuning of the methods. Heron seems to use the lower values for α , β and a perhaps because these hyperparameters act as a regularization value during the parameter learning [3]. Hence the MAP estimation pursued by Heron may require lower regularization values. Finally, we note that the choice K of ranging from 25 to 100 is adequate since the best results occur between this range.

5.5 Summary

In this chapter, we presented the remaining contributions of this thesis. Given the existent trade-off between computational complexity and predictive power of existing topic models, we first proposed a transformation of the Gibbs sampling algorithm that turns the random algorithm into a deterministic version with higher predictive power and easier assessment of convergence. Then, we analyzed the properties of convergence of our proposed algorithm and we find that our algorithm is ultimately finding a fixed point as given by a system of equations. Therefore, we propose an alternative algorithm that aims to solve directly the system of equations. We label this method as Heron inference. The novel proposed algorithm has the advantage of being easily distributable as well as it avoids the computation of repeated tuples. Furthermore, Heron inference can be implemented more efficiently in GPUs because it does not require the use of the limited sampling units available inside GPUs. We prove that our method improves state of the art methods as measured by the learning time, the predictive perplexity and the automated topic coherence. In addition, we show that our method can be used to boost existing application of topic models such as document classification. Throughout the derivation of the proposed method, we find theoretical connections to other inference methods which help explain why our proposed method obtains an improved predictive power. The theoretical connection between several inference methods is also presented in this chapter.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we have presented an study on the assumptions of topic models and their inference methods. In particular we studied how some of the assumptions of topic models have been relaxed to adapt topic models for particular applications and mining of novel types of information. As a main contribution, we have shown how combining an LDA and its dual into a model called dual topics to bicluster (DT2B) yields a novel representation of the co-occurrence relationship between two types of discrete random variables. DT2B generalizes the modeling of LDA by capturing similarity between documents as represented by document-topics. We also modeled the inter-relation between the document-topics and word-topics with an ad hoc probability distribution. This inter-relation is used to extract heavy biclusters in count data. We apply our proposed model to microarray, location and synthetic data to show that the quality of the heavy biclusters extracted by DT2B is better than those extracted by existing biclustering methods. In addition, DT2B is a topic model, hence we showed that the information extracted by the model may be used to improve recommendation

applications such as reviewer recommendation. The source code of our model can be found at <https://github.com/danrugeles/DT2B>.

With respect to the learning of topic models, we studied two families of inference algorithms branching from probabilistic approaches and optimization approaches. The trade-off between the two approaches is given by the fact that probabilistic approaches yield a higher predictive power, while the optimization approaches present approximation algorithms that sacrifice the predictive power for computational efficiency. Several algorithms combine both approaches and they reside in between this trade-off. Ideally a better inference algorithm should improve in both terms of predictive power and computational efficiency. We achieved to do that as follows:

First, we studied that Gibbs sampling follows a point estimation of the substitution sampling method. One advantage of the substitution method is that it is a deterministic algorithm hence its convergence is easily assessed and its benefits translate into reduced computational cost of convergence checks and faster development. Although, the substitution method is more desirable than Gibbs sampling, it is analytically intractable for topic models. Therefore, we proposed to modify topic models to apply a transformation that maps the topic assignment into a soft-topic assignment. Then, the resulting model, labeled as recursive compound allocation (RCA), may use the substitution method for the inference. Beyond determinism, we show empirically that RCA improves the predictive power of collapse Gibbs sampling because the soft-topic assignment follows an improved exploration of the space when compared to the single topic assignment. In addition, we show that RCA is algorithmically equivalent to variational inference for an appropriate selection of a variational family. To the best of our knowledge, this is the first mathematical connection between probabilistic approaches and optimization approaches. In short, RCA contributes by applying a transformation that empowers LDA with soft-topic assignments and enables the

substitution method for LDA. The transformation's result empirically translates in an improved predictive power and a trivial assessment of convergence.

From a different branch, we studied the state augmentation method for MAP estimation which relies on taking several estimates at each step of the Gibbs chain as opposed to the single estimate of Gibbs sampling. The state of the art algorithm within this branch (SAME) presented how the predictive power improves as the number of samples increases. However, SAME can only handle up to about one hundred samples without affecting the computational complexity of the algorithm. As a main contribution of this thesis, we have presented a new inferential method that maximizes the number of samples to infinity and then, using the law large numbers and studying the properties of convergence, we find that the method is ultimately solving a non-linear system of equations. This is beneficial, since the fixed-point iterative algorithm is a well studied algorithm that can solve a non-linear system of equations. We label the application of the fixed point iterative algorithm to the obtained system of equations as Heron inference. Heron inference allows the inference of a family of topic models to be distributable over as many processes as the number of observed records in the dataset. Aside from Heron, all inference methods within the Gibbs sampling family follow an specific ordering which is non trivially distributable. Although, there exists algorithms that solve the Gibbs sampling via distributed algorithms, they sacrifice the predictive power of the model. Instead, Heron always yields the same result independent of the number of cores used in the computation. Furthermore, we show that when the number of samples approaches infinity, the method is equivalent to the substitution method labeled as RCA. Hence, the contribution of Heron to this thesis corresponds to the mathematical manipulation that leads to the system of equations which yield a distributable algorithm. In addition, the system of equations is used to prove that Heron does not need to compute repeated tuples and finally,

since no sampling is required, Heron can also reduce the learning process to the pure computation of arithmetic operations which result in our method running dramatically faster than the state-of-the-art GPU implementation within Gibbs framework. The consistent improvements obtained in computational complexity and predictive power make Heron a compelling algorithm to be used for the inference of topic models in real world applications. We provide an example by implementing and evaluating Heron, SAME and collapsed Gibbs sampling in the well known document classification task. Both GPU and CPU implementations of Heron inference method are available at <https://github.com/danrugeles/Heron/>.

6.2 Future Work

6.2.1 Leveraging Topic Models for Biclustering

In Chapter 4, We have presented how we can use the components of the latent Dirichlet allocation to implement a model which is able to model two different types of topics and their correlation. It is suggested to study how some of the improvements done on topic models may be used to improve the dual topic model. For example, Whye et al. proposed to replace the Dirichlet distribution that models per document-topic distribution with a Dirichlet process that is able to extract the same information as LDA but without having to assume the number of topics [10]. Following the same reasoning, we may improve the proposed Dual topic model so that we don't have to assume the number of topics a priori.

Additionally, there has been interest in extending topic models to include additional information that may improved the predicted power of the model. For example, Harvey et al. incorporated user's search history to improve the document recommendation task using a personalized approach [50]. The model labeled as Personalized Topic

Fig. 6.1 The plate diagram for PTM. Figure from Harvey et al. [50]

Fig. 6.2 PTM model including the inter-relation between user-topics and word-topics.

Model (PTM) is presented in Figure 6.1. This model extended LDA by adding user topics ψ_z in addition to the existing topics of words ϕ_z . The word, user triples w_i, u_i are assigned with a single topic z_i for document d_i . In this model, the topics of users and the topics of words are independent and the correlation between the topics is not extracted. Intuitively mining such correlation may improve the recommendation tasks by finding information about 'which groups of users are interested on querying about which topics'.

As presented in Chapter 4, the presented dual topics to bicluster model introduces the modeling of the dependency between two types of topics which can mine the mutual relationship between the user-topics and the word-topics. To achieve this goal, we may propose to extend PTM to include the user-topics as ψ_k , and then use the parameter Θ to model the relationship between the user-topics and the word-topics. We note that although PTM includes the modeling of user-topics using the parameters ψ_z , the topic assignment is constrained to be one value for an observed word, user pair. Instead, in our proposed model, users and words have its own dedicated topic assignment k_i

and z_i respectively [90]. Figure 6.2 illustrates the graphical model for our proposed extension to PTM.

PTM is one example where two types of topics have been used in the architecture of a model. Several other models have extended topic models to include more types of topics. For instance, Yin et al. extended topic models to incorporate user location using location topics [112]. As in PTM, the correlation between the two types of topics has not been incorporated. As future work, we can use the correlation between topics as an improvement for existing topic models or to design the architecture of new topic models.

6.2.2 Improving the Inference of Topic Models

In Section 5.2, we have presented how the convergence of the Gibbs sampling algorithm is equivalent to finding the fixed points of a system of non linear equations. A fixed point is a value of $\mathbf{x}$ that satisfies Equation 6.1. Numerous problems can be solved by finding fixed-points. One famous example is the solution of Hamilton-Jacobi equations (HJE). HJE present the only alternative formulation of classical mechanics in which the motion of a particle can be represented as a wave. Hence, HJE has fulfilled the goal of theoretical physics of finding the analogy between the propagation of light and the motion of a particle [91].

$$f(\mathbf{x}) = \mathbf{x} \tag{6.1}$$

The most common algorithm used to find fixed points is presented in Algorithm 7. The simplicity of the algorithm allows its application to the solution of many problems including the solution to HJE, or the inference of topic models.

As future work, we could study if we could adapt new approaches to find fixed points in other domains. For example, the one presented by Zhang et al. with the aim of solving HJE [116]. Alternatively, we could implement known optimizations to fixed

point iterators such as the one presented by Todd [99]. The theory about fixed points has grown to the point that Springer has a dedicated journal to the theory of fixed point and applications.

Another promising approach to improve the inference of topic models is to manipulate the system of equations proposing an explicit optimization problem which may be solved using optimization solvers such as Stanford University CVX software [44]. For instance, Equation 6.1 can be reformulated as the minimization problem presented in Equation 6.2.

$$\begin{array}{ll}
 \underset{x}{\text{minimize}} & f(\mathbf{x}) - \mathbf{x} \\
 \text{subject to} & 0 < x_i < 1 \ \forall i \in 1, \dots, K \wedge |\mathbf{x}| = 1.
 \end{array} \tag{6.2}$$

Finally, we may be close to derive a neural network representation equivalent to the topic model representation. The advantage of probabilistic models with respect to neural networks is high interpretability. While neural networks typically consists of layers of weights that map an input to an output. Probabilistic models use probability distributions to represent the random events that lead to the observed data. A connection between a topic model and neural networks may reveal more about how neural networks work. In addition, a representation of topic models using neural networks would enable using topic models as part of more complex neural networks in an end-to-end network.

We may define the neural network equivalent representation to a topic model as the neural network whose trained parameters are theoretically equivalent to the parameters learned using the topic models. We now present one way in which such connection could be found.

We use the Equation 6.1 as the equation whose solution given by $\mathbf{x}$ yields the value of the parameters of the topic model.

For the neural network, we use $h(x) = y$ as the network that transforms x into y . To learn such mapping, we use the cost function $g(x) = \text{cost}(y, y_e)$, where y_e corresponds to the expected output of the network and y corresponds to the output given to the input x in the training dataset. At convergence, we expect $h(x) = y \approx y_e$

We start our connection by transforming Equation 6.1 into a representation that includes the cost function $g(x)$ of a neural network and whose solution remains the same as the solution to Heron inference. For a given constant $\alpha > 0$, we present the transformation in Equation 6.3.

$$f(x) = x' = x - \alpha \nabla g(x) \quad (6.3)$$

Interestingly, Equation 6.3 corresponds to the gradient descent iteration used for minimizing the cost function of a neural network and whose convergence yields the trained parameters of the neural network.

We may now show that at convergence Equation 6.1 and Equation 6.3 are equivalent.

On one hand, Equation 6.3 minimizes the cost function hence, at convergence the sufficient and necessary condition is that $\nabla g(x) \rightarrow 0$. Then, the iteration converges to $f(x) = x$ which is the fixed point iterator obtained for topic models using Heron Inference.

On the other hand, we know that topic convergence occurs when we find x such that $f(x) = x$. Hence, let x be a fixed point of $f(x)$. Then :

$$\begin{aligned}
\nabla g(x) &= \frac{1}{\alpha}(x - (x - \alpha \nabla f(x))) \\
&= \frac{1}{\alpha}(x - f(x)) \\
&= 0
\end{aligned} \tag{6.4}$$

Which shows that the parameters learned by the topic models must converge to those parameters that minimize the cost function of a neural network. However, the connection between neural networks and topic models is still not clear because the cost function $g(x)$ is a function of the error measured between the expected output of the network and the actual output of the network. Therefore, we need to assume an error metric in order to derive the function $h(x)$ that represents the neural network representation of topic model. As future work, we may integrate $\nabla g(x)$ and select an appropriate cost function to reveal the neural network whose solution is guaranteed to be the same as the topic model.

References

- [1] Airoldi, E. M., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2008). Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.*, 9:1981–2014.
- [2] Andrieu, C. and Doucet, A. (2000). Simulated annealing for maximum a posteriori parameter estimation of hidden markov models. *IEEE Transactions on Information Theory*, 46(3):994–1004.
- [3] Asuncion, A., Welling, M., Smyth, P., and Teh, Y. W. (2009). On smoothing and inference for topic models. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pages 27–34, Arlington, Virginia, United States. AUAI Press.
- [4] Banerjee, A., Krumpelman, C., Basu, S., Mooney, R. J., and Ghosh, J. (2005). Model-based overlapping clustering. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-05)*.
- [5] Bao, J., Zheng, Y., and Mokbel, M. F. (2012). Location-based and preference-aware recommendation using sparse geo-social networking data. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '12, pages 199–208, New York, NY, USA. ACM.
- [6] Bauer, S., Noulas, A., Seaghdha, D. O., Clark, S., and Mascolo, C. (2012). Talking places: Modelling and analysing linguistic content in foursquare. In *SocialCom/PASSAT*, pages 348–357. IEEE.
- [7] Baum, L. E. and Petrie, T. (1966). Statistical inference for probabilistic functions of finite state Markov chains. *Annals of Mathematical Statistics*, 37:1554–1563.
- [8] Bertsekas, D. P. and Tsitsiklis, J. N. (1991). Some aspects of parallel and distributed iterative algorithms - A survey' . *Automatica*, 27(1):3–21.
- [9] Blei, D. M. (2012). Probabilistic topic models. *Commun. ACM*, 55(4):77–84.
- [10] Blei, D. M., Griffiths, T. L., and Jordan, M. I. (2010). The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *J. ACM*, 57(2):7:1–7:30.
- [11] Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

- [12] Blei, D. M. and Lafferty, J. D. (2006). Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning, ICML '06*, pages 113–120, New York, NY, USA. ACM.
- [13] Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022.
- [14] Bobadilla, J., Ortega, F., Hernando, A., and Gutiérrez, A. (2013). Recommender systems survey. *Know.-Based Syst.*, 46:109–132.
- [15] Brooks, S. P. and Roberts, G. O. (1998). Convergence assessment techniques for markov chain monte carlo. *Statistics and Computing*, 8(4):319–335.
- [16] Buntine, W. L. and Mishra, S. (2014). Experiments with non-parametric topic models. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14*, pages 881–890, New York, NY, USA. ACM.
- [17] Cao, L. and Fei-Fei, L. (2007). Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. *IEEE 11th International Conference on Computer Vision*, pages 1–8.
- [18] Carpenter, B. (Online; Accessed: 2018-07-02). Integrating out multinomial parameters in latent dirichlet allocation and naive bayes for collapsed gibbs sampling. <https://lingpipe.files.wordpress.com/2010/07/lda3.pdf>.
- [19] Casella, G. and George, E. I. (1992). Explaining the gibbs sampler. *The American Statistician*, 46(3):167–174.
- [20] Chandra, B., Shanker, S., and Mishra, S. (2006). A new approach: Interrelated two-way clustering of gene expression data. *Statistical Methodology*. Bioinformatics.
- [21] Chang, J. (Online; Accessed: 2018-07-02). Derivation of gibbs sampling equations. <https://lists.cs.princeton.edu/pipermail/topic-models/attachments/20110210/89b1646c/attachment-0001.pdf>.
- [22] Chang, J. and Blei, D. M. (2009). Relational topic models for document networks. In Dyk, D. A. V. and Welling, M., editors, *AISTATS*, volume 5 of *JMLR Proceedings*, pages 81–88. JMLR.org.
- [23] Chen, X., Hu, X., Lim, T. Y., Shen, X., Park, E. K., and Rosen, G. L. (2012). Exploiting the functional and taxonomic structure of genomic data by probabilistic topic modeling. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 9(4):980–991.
- [24] Cheng, Y. and Church, G. M. (2000). Biclustering of expression data. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 93–103. AAAI Press.
- [25] Collins, M., Dasgupta, S., and Schapire, R. E. (2002). A generalization of principal components analysis to the exponential family. In Dietterich, T., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 617–624. MIT Press.

- [26] Cook, S. R., Gelman, A., and Rubin, D. B. (2006). Validation of software for bayesian models using posterior quantiles. *Journal of Computational and Graphical Statistics*, 15(3):675–692.
- [27] De Smet, W., Tang, J., and Moens, M.-F. (2011). Knowledge transfer across multilingual corpora via latent topics. In Huang, J. Z., Cao, L., and Srivastava, J., editors, *Advances in Knowledge Discovery and Data Mining*, pages 549–560, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [28] Dhillon, I. S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 269–274, New York, NY, USA. ACM.
- [29] Dhillon, I. S., Mallela, S., and Modha, D. S. (2003). Information-theoretic co-clustering. KDD.
- [30] Doucet, A., Godsill, S. J., and Robert, C. P. (2002). Marginal maximum a posteriori estimation using markov chain monte carlo. *Statistics and Computing*, 12(1):77–84.
- [31] Du, L., Buntine, W., and Jin, H. (2010). Sequential latent dirichlet allocation: Discover underlying topic structures within a document. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 148–157.
- [32] Eren, K., Deveci, M., Küçüktunç, O., and Çatalyürek, Ü. V. (2013). A comparative analysis of biclustering algorithms for gene expression data. *Briefings in Bioinformatics*, 14(3):279–292.
- [33] Escobar, M. D. and West, M. (1994). Bayesian density estimation and inference using mixtures. *Journal of the American Statistical Association*, 90:577–588.
- [34] Falcon, S. and Gentleman, R. (2007). Using gostats to test gene lists for go term association. *Bioinformatics*, 23(2):257–258.
- [35] Farrahi, K. and Gatica-Perez, D. (2011). Discovering routines from large-scale human locations using probabilistic topic models. *ACM Trans. Intell. Syst. Technol.*, 2(1):3:1–3:27.
- [36] Fox, C. and Roberts, S. (2012). A tutorial on variational Bayesian inference. *Artificial Intelligence Review*, 38(2):85–95.
- [37] Fox, E., Sudderth, E., Jordan, M., and Willsky, A. (2011). Bayesian nonparametric inference of switching dynamic linear models. *Signal Processing, IEEE Transactions on*, 59(4):1569–1585.
- [38] Fox, E. B., Sudderth, E. B., Jordan, M. I., and Willsky, A. S. (2008). An hdp-hmm for systems with state persistence. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 312–319, New York, NY, USA. ACM.

- [39] Gan, X., Liew, A. W., and Yan, H. (2008). Discovering biclusters in gene expression data based on high-dimensional linear geometries. *BMC Bioinformatics*, 9.
- [40] Gelfand, A. E. and Smith, A. F. M. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409.
- [41] Getz, G., Levine, E., and Domany, E. (2000). Coupled two-way clustering analysis of gene microarray data. *Proceedings of the National Academy of Sciences*, (22):12079–12084.
- [42] Ghahramani, Z. and Beal, M. J. (2000). Variational inference for bayesian mixtures of factor analysers. In *In Advances in Neural Information Processing Systems 12*, pages 449–455. MIT Press.
- [43] Ghahramani, Z. and Jordan, M. I. (1997). Factorial hidden markov models. *Mach. Learn.*, 29(2-3):245–273.
- [44] Grant, M. and Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. <http://cvxr.com/cvx>.
- [45] Griffiths, T. L. and Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235.
- [46] Griffiths, T. L., Steyvers, M., Blei, D. M., and Tenenbaum, J. B. (2005). Integrating topics and syntax. In Saul, L., Weiss, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 17*, pages 537–544. MIT Press.
- [47] Gruber, A., Weiss, Y., and Rosen-Zvi, M. (2007). Hidden topic markov models. In *AISTATS*, volume 2 of *JMLR Proceedings*, pages 163–170. JMLR.org.
- [48] Hai, Z., Cong, G., Chang, K., Cheng, P., and Miao, C. (2017). Analyzing sentiments in one go: A supervised joint topic modeling approach. *IEEE Trans. on Knowl. and Data Eng.*, 29(6):1172–1185.
- [49] Hartigan, J. A. (1972). Direct clustering of a data matrix. 67(337):123–129.
- [50] Harvey, M., Crestani, F., and Carman, M. J. (2013). Building user profiles from topic models for personalised search. In *Proceedings of the 22Nd ACM International Conference on Information & Knowledge Management, CIKM '13*, pages 2309–2314, New York, NY, USA. ACM.
- [51] He, B. D., Sa, C. D., Mitliagkas, I., and Ré, C. (2016). Scan order in gibbs sampling: Models in which it matters and bounds on how much. *CoRR*, abs/1606.03432.
- [52] Hoffman, M., Bach, F. R., and Blei, D. M. (2010). Online learning for latent dirichlet allocation. In Lafferty, J. D., Williams, C. K. I., Shawe-Taylor, J., Zemel, R. S., and Culotta, A., editors, *Advances in Neural Information Processing Systems 23*, pages 856–864. Curran Associates, Inc.
- [53] Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *J. Mach. Learn. Res.*, 14(1):1303–1347.

- [54] Hofmann, T. (1999). Probabilistic latent semantic analysis. In *In Proc. of Uncertainty in Artificial Intelligence, UAI'99*, pages 289–296.
- [55] Hu, D. and Saul, L. K. (2009). A probabilistic topic model for unsupervised learning of musical key-profiles. In Hirata, K., Tzanetakis, G., and Yoshii, K., editors, *ISMIR*, pages 441–446. International Society for Music Information Retrieval.
- [56] Ihmels, J., Bergmann, S., and Barkai, N. (2004). Defining transcription modules using large-scale gene expression data. *Bioinformatics*, 20:2004.
- [57] Ishwaran, H. and James, L. F. (2001). Gibbs Sampling Methods for Stick-Breaking Priors. *Journal of the American Statistical Association*, 96(453):161–173.
- [58] Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME – Journal of Basic Engineering*, (82 (Series D)):35–45.
- [59] Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. M. (2017). Automatic differentiation variational inference. *J. Mach. Learn. Res.*, 18(1):430–474.
- [60] Lafferty, J. D. and Blei, D. M. (2006). Correlated topic models. In Weiss, Y., Schölkopf, B., and Platt, J. C., editors, *Advances in Neural Information Processing Systems 18*, pages 147–154. MIT Press.
- [61] Landauer, T., Mcnamara, D., Dennis, S., and Kintsch, W. (2007). *Handbook of Latent Semantic Analysis*. Lawrence Erlbaum Associates.
- [62] Lang, K. (1995). Newsweeder: Learning to filter netnews. In *in Proceedings of the 12th International Machine Learning Conference (ML95)*.
- [63] Lau, J. H., Newman, D., and Baldwin, T. (2014a). 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, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 530–539.
- [64] Lau, J. H., Newman, D., and Baldwin, T. (2014b). Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *EACL*, pages 530–539. The Association for Computer Linguistics.
- [65] Lazzeroni, L. and Owen, A. (2000). Plaid models for gene expression data. *Statistica Sinica*, 12:61–86.
- [66] Li, W. and McCallum, A. (2006). Pachinko allocation: Dag-structured mixture models of topic correlations. In *Proceedings of the 23rd International Conference on Machine Learning, ICML '06*, pages 577–584, New York, NY, USA. ACM.
- [67] Liu, B., Fu, Y., Yao, Z., and Xiong, H. (2013). Learning geographical preferences for point-of-interest recommendation. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13*, pages 1043–1051, New York, NY, USA. ACM.

- [68] Liu, B. and Xiong, H. (2013). Point-of-interest recommendation in location based social networks with topic and location awareness. In *Proceedings of the 13th SIAM International Conference on Data Mining, May 2-4, 2013. Austin, Texas, USA.*, pages 396–404.
- [69] Ma, Q. M., Tang, H., Paterson, A., and Ying, X. (2009). Qubic: a qualitative biclustering algorithm for analyses of gene expression data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*.
- [70] Madeira, S. and Oliveira, A. (2004). Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):24–45.
- [71] Mahfouz, M. and Ismail, M. (2012). Soft flexible overlapping biclustering utilizing hybrid search strategies. In Hassanien, A., Salem, A.-B., Ramadan, R., and Kim, T.-h., editors, *Advanced Machine Learning Technologies and Applications*, volume 322 of *Communications in Computer and Information Science*, pages 315–326. Springer Berlin Heidelberg.
- [72] Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA.
- [73] Mcauliffe, J. D. and Blei, D. M. (2008). Supervised topic models. In Platt, J. C., Koller, D., Singer, Y., and Roweis, S. T., editors, *Advances in Neural Information Processing Systems 20*, pages 121–128. Curran Associates, Inc.
- [74] Mei, Q., Liu, C., Su, H., and Zhai, C. (2006). A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *Proceedings of the 15th International Conference on World Wide Web, WWW '06*, pages 533–542, New York, NY, USA. ACM.
- [75] Minka, T. and Lafferty, J. (2002). Expectation-propagation for the generative aspect model. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence, UAI'02*, pages 352–359, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- [76] Murali, T. M. and Kasif, S. (2003). Extracting conserved gene expression motifs from gene expression data. In *Pac. Symp. Biocomput*, pages 77–88.
- [77] Newman, D., Lau, J. H., Grieser, K., and Baldwin, T. (2010). Automatic evaluation of topic coherence. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10*, pages 100–108, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [78] Paisley, J., Blei, D. M., and Jordan, M. I. (2012). Variational bayesian inference with stochastic search. In *Proceedings of the 29th International Conference on Machine Learning, ICML'12*, pages 1363–1370, USA. Omnipress.

- [79] Paisley, J. and Carin, L. (2009). Nonparametric factor analysis with beta process priors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 777–784, New York, NY, USA. ACM.
- [80] Perrone, V., Jenkins, P. A., Spano, D., and Teh, Y. W. (2017). Poisson random fields for dynamic feature models. *Journal of Machine Learning Research*, 18:1–45.
- [81] Pontes, B., Giráldez, R., and Aguilar-Ruiz, J. S. (2015a). Biclustering on expression data: A review. *Journal of Biomedical Informatics*, 57:163 – 180.
- [82] Pontes, Beatriz, G., Raúl, and Aguilar-Ruiz, J. S. (2015b). Biclustering on expression data: A review. *Journal of Biomedical Informatics*.
- [83] Prelić, A., Bleuler, S., Zimmermann, P., Wille, A., Bühlmann, P., Gruissem, W., Hennig, L., Thiele, L., and Zitzler, E. (2006). A systematic comparison and evaluation of biclustering methods for gene expression data. *Bioinformatics*, 22(9):1122–1129.
- [84] Ramage, D., Hall, D., Nallapati, R., and Manning, C. D. (2009). Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 - Volume 1*, EMNLP '09, pages 248–256, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [85] Ranganath, R., Gerrish, S., and Blei, D. M. (2014). Black box variational inference. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics, AISTATS 2014, Reykjavik, Iceland, April 22-25, 2014*, pages 814–822.
- [86] Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407.
- [87] Rosen-Zvi, M., Griffiths, T., Steyvers, M., and Smyth, P. (2004). The author-topic model for authors and documents. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, UAI '04, pages 487–494, Arlington, Virginia, United States. AUAI Press.
- [88] Roy, D. M. and Teh, Y. W. (2009). The mondrian process. In Koller, D., Schuurmans, D., Bengio, Y., and Bottou, L., editors, *Advances in Neural Information Processing Systems 21*, pages 1377–1384.
- [89] Rugeles, D., Hai, Z., Cong, G., and Dash, M. (2018). Heron inference for bayesian graphical models. *CoRR*, abs/1802.06526.
- [90] Rugeles, D., Zhao, K., Cong, G., Dash, M., and Krishnaswamy, S. (2017). Biclustering: An application of dual topic models. In *Proceedings of the 2017 SIAM International Conference on Data Mining, Houston, Texas, USA, April 27-29, 2017.*, pages 453–461.
- [91] Sakurai, J. J. (1994). *Modern quantum mechanics; rev. ed.* Addison-Wesley, Reading, MA.
- [92] Shan, H. and Banerjee, A. (2008). Bayesian co-clustering. In *Data Mining, 2008. ICDM '08. Eighth IEEE International Conference on*, pages 530–539.

- [93] Smola, A. and Narayanamurthy, S. (2010). An architecture for parallel topic models. *Proc. VLDB Endow.*, 3(1-2):703–710.
- [94] Sontag, D. and Roy, D. (2011). Complexity of inference in latent dirichlet allocation. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 24*, pages 1008–1016. Curran Associates, Inc.
- [95] Steyvers, M. and Griffiths, T. (2007). *Probabilistic Topic Models*. Lawrence Erlbaum Associates.
- [96] Tanay, A., Sharan, R., and Shamir, R. (2002). Discovering statistically significant biclusters in gene expression data. In *ISMB*, pages 136–144.
- [97] Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. (2004). Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101.
- [98] Teh, Y. W., Newman, D., and Welling, M. (2007). A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *Advances in Neural Information Processing Systems 19*, pages 1353–1360. MIT Press.
- [99] Todd, M. J. (1978). *Improving the convergence of fixed-point algorithms*, pages 151–169. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [100] Walker, H. F. and Ni, P. (2011). Anderson acceleration for fixed-point iterations. *SIAM J. Numer. Anal.*, 49(4):1715–1735.
- [101] Wallach, H. M. (2006). Topic modeling: Beyond bag-of-words. In *Proceedings of the 23rd International Conference on Machine Learning, ICML '06*, pages 977–984, New York, NY, USA. ACM.
- [102] Wallach, H. M., Murray, I., Salakhutdinov, R., and Mimno, D. (2009). Evaluation methods for topic models. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML '09*, pages 1105–1112, New York, NY, USA. ACM.
- [103] Wang, C., Blei, D. M., and Heckerman, D. (2008). Continuous time dynamic topic models. In McAllester, D. A. and Myllymäki, P., editors, *UAI*, pages 579–586. AUAI Press.
- [104] Wang, X. and McCallum, A. (2006). Topics over time: A non-markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '06*, pages 424–433, New York, NY, USA. ACM.
- [105] Wang, X., McCallum, A., and Wei, X. (2007). Topical n-grams: Phrase and topic discovery, with an application to information retrieval. In *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining, ICDM '07*, pages 697–702, Washington, DC, USA. IEEE Computer Society.

- [106] Wei, X., Sun, J., and Wang, X. (2007). Dynamic mixture models for multiple time series. In *Proceedings of the International Joint Conference on Artificial Intelligence*, volume 459, pages 2909–2914.
- [107] Welling, M., Teh, Y. W., and Kappen, B. (2008). Hybrid variational/gibbs collapsed inference in topic models. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, UAI’08, pages 587–594, Arlington, Virginia, United States. AUAI Press.
- [108] Yan, F., Xu, N., and Qi, Y. A. (2009). Parallel inference for latent dirichlet allocation on graphics processing units. In *Proceedings of the 22Nd International Conference on Neural Information Processing Systems*, NIPS’09, pages 2134–2142, USA. Curran Associates Inc.
- [109] Yan, X., Guo, J., Lan, Y., and Cheng, X. (2013). A biterm topic model for short texts. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW ’13, pages 1445–1456, New York, NY, USA. ACM.
- [110] Yang, D., Zhang, D., Zheng, V., and Yu, Z. (2015). Modeling user activity preference by leveraging user spatial temporal characteristics in lbsns. *Systems, Man, and Cybernetics: Systems, IEEE Transactions on*, 45(1):129–142.
- [111] Yin, Z., Cao, L., Han, J., Zhai, C., and Huang, T. (2011a). Geographical topic discovery and comparison. In *Proceedings of the 20th International Conference on World Wide Web*, WWW ’11, pages 247–256, New York, NY, USA. ACM.
- [112] Yin, Z., Cao, L., Han, J., Zhai, C., and Huang, T. (2011b). Geographical topic discovery and comparison. In *Proceedings of the 20th International Conference on World Wide Web*, WWW ’11, pages 247–256, New York, NY, USA. ACM.
- [113] Yuan, Q., Cong, G., Ma, Z., Sun, A., and Thalmann, N. M. (2013). Who, where, when and what: Discover spatio-temporal topics for twitter users. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’13, pages 605–613, New York, NY, USA. ACM.
- [114] Zellner, A. and Min, C.-K. (1995). Gibbs sampler convergence criteria. *Journal of the American Statistical Association*, 90(431):921–927.
- [115] Zhang, J., Song, Y., Zhang, C., and Liu, S. (2010). Evolutionary hierarchical dirichlet processes for multiple correlated time-varying corpora. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’10, pages 1079–1088, New York, NY, USA. ACM.
- [116] Zhang, Y.-T., Zhao, H.-K., and Chen, S. (2006). Fixed-point iterative sweeping methods for static hamilton-jacobi equations. *Methods Appl. Anal.*, 13(3):299–320.
- [117] Zhao, H., Jiang, B., Canny, J. F., and Jaros, B. (2015). Same but different: Fast and high quality gibbs parameter estimation. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’15, pages 1495–1502, New York, NY, USA. ACM.
- [118] Zhou, S., Li, K., and Liu, Y. (2009). Text categorization based on topic model. *International Journal of Computational Intelligence Systems*, 2(4):398–409.

Appendix A

Inference of DT2B via Collapsed Gibbs Sampling

In this section we derive the posterior distribution used to assign the topics. The derivation includes two main sections. First, we apply collapsed Gibbs sampling to derive analytically the posterior distribution given the conditions of the model. Second, we use the fact that every record is independent and identically distributed given θ and ϕ so that we can find the posterior derivation for each particular record in the dataset.

A.1 Posterior derivation given the conditions of the model

First we write down the target distribution that we want to derive:

$$p(z_{r,v}, z_{c,v} | z_{r,-v}, z_{c,-v}, u, p, \alpha, \beta_r, \beta_c) \tag{A.1}$$

The idea is to find the updating topics z_c and z_r for a given visit record v , given the training data.

Now, we start by formulating the joint distribution over all the variables in the model.

$$p(r, p, z_r, z_c, \phi_r, \phi_c, \theta | \alpha, \beta_r, \beta_c) = \quad (\text{A.2})$$

To obtain the target distribution using collapsed Gibbs sampling, we need to integrate out the unknown parameters ϕ_r, ϕ_c and θ .

$$\int_{\theta} \int_{\phi_r} \int_{\phi_c} p(r, p, z_r, z_c, \phi_r, \phi_c, \theta | \alpha, \beta_r, \beta_c) d\phi_c d\phi_r d\theta = \quad (\text{A.3})$$

Next, we apply DT2B's independence assumptions.

$$\int_{\theta} \int_{\phi_r} \int_{\phi_c} p(\theta | \alpha) p(z_c | \theta) p(z_r | \theta) p(c | \phi_c) p(\phi_c | \beta_c) p(r | \phi_r) p(\phi_r | \beta_r) d\phi_c d\phi_r d\theta \quad (\text{A.4})$$

We split the integrals into three independent cases:

Case 1 (θ):

$$\int_{\theta} p(\theta | \alpha) p(z_c | \theta) p(z_r | \theta) d\theta = \quad (\text{A.5})$$

We use the template to describe the V number of visit records:

$$\begin{aligned}
\int_{\theta} p(\theta|\alpha) \prod_{v=1}^V (c(z_{c,v}|\theta) p(z_{r,v}|\theta)) d\theta = \\
\int_{\theta} p(\theta|\alpha) \prod_{v=1}^V (c(z_{c,v}, z_{r,v}|\theta)) d\theta =
\end{aligned} \tag{A.6}$$

θ is a $K_r x K_c$ dimensional variable. As demonstrated in Section 4.3.3, θ_k can be transformed into $\theta_{i,j}$ using a bijection function g where $i \in \mathbb{N}_{1:K_u}, j \in \mathbb{N}_{1:K_p}$.

$$\int_{\theta_k} p(\theta_k|\alpha) \prod_{v=1}^V (c(z_{c,v}, z_{r,v}|\theta_{i,j})) d\theta_k = \tag{A.7}$$

We replace the probability distribution with the definition of the Dirichlet distribution with equal number of parameters α as the dimensionality of θ .

$$\iint_{\theta_{i,j}} \frac{\Gamma\left(\sum_{k=1}^{K_r x K_c} (\alpha_k)\right)}{\prod_{k=1}^{K_r x K_c} \Gamma(\alpha_k)} \prod_{k=1}^{K_r x K_c} (\theta_k^{\alpha_k-1}) \prod_{v=1}^V (\theta_{i,j}) d\theta_{i,j} = \tag{A.8}$$

For simplicity, We define the counting function $c(a, b, c, d)$ using an SQL query:

```

Select  count(*)
From    VisitRecords
Where   usertopic=a
        and placetopic=b
        and user=c
        and place=d

```

If a, b, c or d are equal to $*$, then the correspondent filter is removed from the *Where* clause.

$$\int_{\theta_{i,j}} \frac{\Gamma\left(\sum_{K_r, K_c=1}^{K_r x K_c} (\alpha_{ku, kp})\right)}{\prod_{K_r, K_c=1}^{K_r x K_c} (\Gamma(\alpha_{ku, kp}))} \prod_{K_r, K_c=1}^{K_r x K_c} \left(\theta_{ku, kp}^{\alpha_{ku, kp}-1}\right) \prod_{K_r, K_c=1}^{K_r x K_c} \left(\theta_{ku, kp}^{c(K_c, K_r, *, *)}\right) d\theta_{kp} \propto \quad (\text{A.9})$$

Now, we arrange the variables so that we can describe a Dirichlet distribution which will integrate to one. This method has been used to derive the updating topics LDA¹. After the integration we find the following:

$$\frac{\prod_{K_r, K_c=1}^{K_r x K_c} (\Gamma(c(K_c, K_r, *, *) + \alpha_{ku, kp}))}{\Gamma\left(\sum_{K_r, K_c=1}^{K_r x K_c} (c(K_c, K_r, *, *) + \alpha_{ku, kp})\right)} \quad (\text{A.10})$$

Case 2 (ϕ_r):

Following a similar formulation that the one used for case 1.

$$\int_{\phi_r} p(r|\phi_r) p(\phi_r|\beta_r) d\phi_r = \prod_{K_r=1}^{K_r} \left(\int_{\phi_r} p(r|\phi_r) p(\phi_r|\beta_r) d\phi_r \right) \quad (\text{A.11})$$

We obtain:

$$\prod_{K_r=1}^{K_r} \left(\frac{\prod_{v=1}^V (\Gamma(c(*, K_r, *, u_v) + \beta_{u,v}))}{\Gamma\left(\sum_{v=1}^V (c(*, K_r, *, u_v) + \beta_{u,v})\right)} \right) \quad (\text{A.12})$$

Case 3 (ϕ_c):

Following the same formulation that the one used for case 2.

$$\int_{\phi_c} p(c|\phi_c) p(\phi_c|\beta_c) d\phi_c = \prod_{K_c=1}^{K_c} \left(\int_{\phi_c} p(c|\phi_c) p(\phi_c|\beta_c) d\phi_c \right) \quad (\text{A.13})$$

¹<http://lingpipe-blog.com/2010/07/13/collapsed-gibbs-sampling-for-lda-bayesian-naive-bayes/>

We obtain:

$$\prod_{K_c=1}^{K_c} \left(\frac{\prod_{v=1}^V (\Gamma(c(K_c, *, p_v, *) + \beta_{p,v}))}{\Gamma(\sum_{v=1}^V (c(K_c, *, p_v, *) + \beta_{p,v}))} \right) \quad (\text{A.14})$$

In Summary:

$$p(z_r, z_c, u, p | \alpha, \beta_r, \beta_c) \propto$$

$$\frac{\prod_{K_r, K_c=1}^{K_r \times K_c} (\Gamma(c(K_c, K_r, *, *) + \alpha_{ku, kp}))}{\Gamma(\sum_{K_r, K_c=1}^{K_r \times K_c} (c(K_c, K_r, *, *) + \alpha_{ku, kp}))} \prod_{K_r=1}^{K_r} \left(\frac{\prod_{v=1}^V (\Gamma(c(*, K_r, *, u_v) + \beta_{u,v}))}{\Gamma(\sum_{v=1}^V (c(*, K_r, *, u_v) + \beta_{u,v}))} \right) \prod_{K_c=1}^{K_c} \left(\frac{\prod_{v=1}^V (\Gamma(c(K_c, *, p_v, *) + \beta_{p,v}))}{\Gamma(\sum_{v=1}^V (c(K_c, *, p_v, *) + \beta_{p,v}))} \right) \quad (\text{A.15})$$

A.2 Posterior derivation for each record

Since every visit record is considered to be independent, we can update every single record's user-topic and place-topic instead, we refine our objective probability function to the following:

$$p(z_r^{(v)}, z_c^{(v)} | z_r^{(-v)}, z_c^{(-v)}, u_v, p_v, \alpha, \beta_r, \beta_c) \quad (\text{A.16})$$

Where $z_r^{(v)}, z_c^{(v)}$ represents the user and place topic for the v -th record and $z_r^{(-v)}, z_c^{(-v)}$ represent the user and place topic for all other records. Once again, we split equation A.15 into three cases.

Case 1 (θ): Given $u_v, p_v, z_r^{(-v)}, z_c^{(-v)}$:

$$\frac{\prod_{K_r, K_c=1}^{K_r x K_c} \left(\Gamma \left(c(K_c, K_r, *, *)^{-v} + \alpha_{ku, kp} \right) \right)}{\Gamma \left(\sum_{K_r, K_c=1}^{K_r x K_c} \left(c(K_c, K_r, *, *)^{-v} + \alpha_{ku, kp} \right) \right)} \quad (\text{A.17})$$

We use the property of the gamma distribution $\Gamma(x+1) = \Gamma(x)x$ in order to extract the components that correspond to our v -th record of interest.

$$\frac{\prod_{K_r, K_c \neq z_{c,v}, z_{r,v}}^{K_r x K_c} \left(\Gamma \left(c(K_c, K_r, *, *)^{-v} + \alpha_{ku, kp} \right) \Gamma \left(c(z_{c,v}, z_{r,v}, *, *)^{-v} + \alpha_{zu_v, zp_v} \right) \right)}{\Gamma \left(1 + \sum_{K_r, K_c=1}^{K_r x K_c} \left(c(K_c, K_r, *, *)^{-v} + \alpha_{ku, kp} \right) \right)} \frac{c(z_{c,v}, z_{r,v}, *, *)^{-v} + \alpha_{zu_v, zp_v}}{\Gamma \left(1 + \sum_{K_r, K_c=1}^{K_r x K_c} \left(c(K_c, K_r, *, *)^{-v} + \alpha_{ku, kp} \right) \right)} \propto$$

Finally, after simplifying all terms that are not proportional to the topics $z_r^{(v)}, z_c^{(v)}$

$$c(z_{c,v}, z_{r,v}, *, *)^{-v} + \alpha_{zu_v, zp_v} \quad (\text{A.18})$$

We use the same formulation for the other cases and we obtain the updating equation for $z_r^{(v)}, z_c^{(v)}$

$$p \left(z_r^{(v)}, z_c^{(v)} | z_r^{(-v)}, z_c^{(-v)}, u_v, p_v, \alpha, \beta_r, \beta_c \right) \propto$$

$$\frac{\left(c(z_{c(v)}, z_{r(v)}, *, *)^{(-v)} + \alpha_{z_{r(v)}, z_{c(v)}} \right) \left(c(z_{c(v)}, *, p_{(v)}, *)^{(-v)} + \beta_{p(v)} \right) \left(c(*, z_{r(v)}, *, u_{(v)})^{(-v)} + \beta_{u(v)} \right)}{\sum_{j=1}^{|p|} \left(c(z_{c(v)}, *, p_{(j)}, *)^{(-v)} + \beta_{p(j)} \right) \sum_{i=1}^{|U|} \left(c(*, z_{r(v)}, *, u_{(i)})^{(-v)} + \beta_{u(i)} \right)} \quad (\text{A.19})$$

A.3 Perplexity of a topic using DT2B

To compare DT2B's perplexity with an LDA model, we need to extract the user-topics $p(r|\phi_u, \alpha)$ and the place-topics $p(c|\phi_p, \alpha)$ separately. Because these distributions are intractable, we use Chi-Square estimation method [102] to obtain an approximate result. The estimation method requires that we derive analytically $p(r, z_r|\phi_u, \alpha)$, and $p(c, z_c|\phi_p, \alpha)$. We can obtain the former distribution since $p(r, z_r|\phi_u, \alpha) = p(r|\phi_u, z_r)p(z_r|\alpha)$. Also, we can obtain $p(r|\phi_u, z_r)$ from the inference process. With respect to $p(z_r|\alpha)$, we can start by writing out $p(z_r, z_c|\alpha) = \int_{\theta} p(z_r, z_c|\theta)p(\theta|\alpha)$ which can be re-written as $p(z_r, z_c|\alpha) = \int_{\theta_i} \int_{\theta_j} p(z_r|\theta_j)p(\theta_j|\alpha)p(z_c|\theta_i)p(\theta_i|\alpha)$. Finally, we to obtain $p(z_r|\alpha) = \int_{\theta_j} p(z_r|\theta_j)p(\theta_j|\alpha) \propto \frac{\prod_{i=1}^{K_p} \left(\Gamma \left(c(i, z_r, *, *)^{-v} + \alpha_{z_r \times K_p + i} \right) \right)}{\Gamma \left(\sum_{i=1}^{K_u} \left(c(i, K_r, *, *)^{-v} + \alpha_{z_r \times K_p + i} \right) \right)}$