Topic Uncovering and Image Annotation via Scalable Probit Normal Correlated Topic Models

Xingchen Yu

Follow this and additional works at: https://scholarworks.rit.edu/theses

Recommended Citation

This Thesis is brought to you for free and open access by RIT Scholar Works. It has been accepted for inclusion in Theses by an authorized administrator of RIT Scholar Works. For more information, please contact ritscholarworks@rit.edu.
Rochester Institute of Technology

Master’s Thesis

Topic Uncovering and Image Annotation via Scalable Probit Normal Correlated Topic Models

Author: Xingchen Yu
Advisor: Dr. Ernest Fokoué

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Applied Statistics

In the

The John D. Hromi Center for Quality and Applied Statistics
The Kate Gleason College of Engineering

May 2015
Committee Approval

The undersigned have examined thesis titled “Topic Uncovering and Image Annotation via Scalable Probit Normal Correlated Topic Model” by Xingchen Yu, a candidate for the degree of Master of Science in Applied Statistics, and hereby approve that it is worthy of acceptance:

______________________________  ____________________
Dr. Ernest Fokoué                    Date

______________________________  ____________________
Dr. Joseph Voelkel                  Date

______________________________  ____________________
Dr. Daniel Lawrence                Date
Declaration of Authorship

I, Xingchen Yu, declare that this thesis titled, “Topic Uncovering and Image Annotation via Scalable Probit Normal Correlated Topic Model” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given.

With the exception of such quotations, this thesis is entirely my own work.

- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed

Date
“Have the courage to follow your heart and intuition. They somehow know what you truly want to become.”

Steve Jobs
Abstract

The Kate Gleason College of Engineering
Master of Science

Topic Uncovering and Image Annotation via Scalable Probit Normal Correlated Topic Models
by Xingchen Yu

Topic uncovering of the latent topics have become an active research area for more than a decade and continuous to receive contributions from all disciplines including computer science, information science and statistics. Since the introduction of Latent Dirichlet Allocation in 2003, many intriguing extension models have been proposed. One such extension model is the logistic normal correlated topic model, which not only uncovers hidden topic of a document, but also extract a meaningful topical relationship among a large number of topics. In this model, the Logistic normal distribution was adapted via the transformation of multivariate Gaussian variables to model the topical distribution of documents in the presence of correlations among topics. In this thesis, we propose a Probit normal alternative approach to modelling correlated topical structures. Our use of the Probit model in the context of topic discovery is novel, as many authors have so far concentrated solely of the logistic model partly due to the formidable inefficiency of the multinomial Probit model even in the case of very small topical spaces. We herein circumvent the inefficiency of multinomial Probit estimation by using an adaptation of
the Diagonal Orthant Multinomial Probit (DO-Probit) in the topic models context, resulting in the ability of our topic modeling scheme to handle corpuses with a large number of latent topics. In addition, we extended our model and implement it into the context of image annotation by developing an efficient Collapsed Gibbs Sampling scheme. Furthermore, we employed various high performance computing techniques such as memory-aware Map Reduce, SparseLDA implementation, vectorization and block sampling as well as some numerical efficiency strategy to allow fast and efficient sampling of our algorithm.
Acknowledgements

I am especially grateful to my advisor Dr. Fokoué and I would like to express my sincere gratitude for his constant guidance and kind help throughout the completion of this thesis. During the time under his mentorship and stewardship, I redeemed myself and am awarded with many PhD offers. More importantly I made a friend for a lifetime.

I wish to thank Dr. Lawrence and Dr. Voelkel wholeheartedly for their acceptance of being in my thesis committee and their comprehensive and constructive advice. I also would like thank them for the knowledge they have taught me in the past two years.

I wish to thank all the CQAS faculty and staff for teaching and helping me over the course of my study in the department.

I wish to thank the CQAS department for accepting me as a student and offering me the scholarship to support my study for the master’s degree.

I wish to thank my parents for their constant support and belief in me.

I wish to express our heartfelt gratitude and infinite thanks to Our Lady of Perpetual Help for her ever-present support and guidance, especially for the uninterrupted flow of inspiration received through her most powerful intercession.
List of Figures

Figure 1 Pipeline for Natural Language Processing ([10]) ........................................ 8
Figure 2 Probabilistic Graphical Model of LDA ......................................................... 15
Figure 3 Geometrical Representation of a Document with Three Topics ([10]) ............ 17
Figure 4 Symmetric Dirichlet Samples for K=5 ([10]) ............................................. 18
Figure 5 Sample Path of a Gibbs Sampler ................................................................. 23
Figure 6 Correlation Structure between Topics ([2]) ................................................. 28
Figure 7 Probabilistic Graphical Model for CTM ...................................................... 29
Figure 8 Loglikelihood as a Function of Topics ......................................................... 48
Figure 9 Correlation Structures of Topics .................................................................. 50
Figure 10 Example from BBC dataset ...................................................................... 55
Figure 11 Example of Noisy Input ............................................................................ 56
Figure 12 Difference of Gaussian ([41]) .................................................................. 58
Figure 13 Searching of Local Extrema ([41]) ............................................................. 58
Figure 14 Flowchart of Image Clustering ([45]) ......................................................... 60
Figure 15 Probabilistic Graphical Model of Image Annotation via Correlated Topic Model .............................................................................................................. 62
Figure 16 Annotation Results ..................................................................................... 66
Figure 17 Efficiency Comparisons between Parallel and Non-Parallel......................... 73
Figure 18 Pipeline of Map Reduce ............................................................................. 74
Figure 19 Parallel Implementation of LDA [31] .......................................................... 75
List of Tables

Table 1 Example of Document Term Matrix .............................................................. 13
Table 2 Basic Terminology ..................................................................................... 14
Table 3 Efficient Comparison Before and After the Adaption of DO Probit ............ 47
Table 4 Representation of Topics Discoved .......................................................... 49
Table 5 Topic Assignment Matrix for a Document ................................................. 79
# Table of Contents

Committee Approval............................................................................................................. i  
Declaration of Authorship..................................................................................................... ii  
Abstract.............................................................................................................................. iv  
Acknowledgements.............................................................................................................. vi  
List of Figures ....................................................................................................................... vii  
List of Tables ......................................................................................................................... viii  
Table of Contents ................................................................................................................ ix  

1. Introduction and Overview ............................................................................................... 1  
   1.1 Motivation and Scope of Thesis ................................................................................. 2  
   1.2 Thesis Organization ................................................................................................. 4  

2. Thesis Background ............................................................................................................ 5  
   2.1 Text Mining and Natural Language Processing ......................................................... 6  
   2.2 The Original Topic Model - Latent Dirichlet Allocation ............................................. 9  
      2.2.1 Introduction with a Motivating Example .......................................................... 9  
      2.2.2 Model Input and Assumptions ........................................................................ 11  
      Assumptions ................................................................................................................ 11  
      Model Input ................................................................................................................ 12  
      2.2.3 Generative Process of Latent Dirichlet Allocation ......................................... 13  
      Dirichlet Distribution ................................................................................................. 16  
      Parameter Estimation ................................................................................................. 18  
      Model Evaluation Method .......................................................................................... 27  
   2.3 Correlated Topic Model ............................................................................................. 28  
      2.3.1 General Aspects .............................................................................................. 28  
      2.3.2 Parameter Estimation ...................................................................................... 30  

3. Probit Normal Correlated Topic Model .......................................................................... 33  
   3.1 Introduction ............................................................................................................... 33  
   3.2 General Aspects of the Probit normal correlated topic model ................................ 36  
   3.3 Diagonal Orthant Probit for Correlated Topic Models ............................................. 39
3.4 Computational Results on the Associated Press data................................................. 45
3.5 Conclusion and Discussion ......................................................................................... 51
4. Image Annotation via the Probit Normal Correlated Topic Model............................... 53
   4.1 Introduction and Related Works .............................................................................. 53
   4.2 Model Input and Data Description .......................................................................... 54
   4.3 Scale Invariant Feature Transform (SIFT) .............................................................. 57
   4.4 Generative Process .................................................................................................. 60
   4.5 Parameter Estimation .............................................................................................. 62
   4.6 Computational Results ............................................................................................ 66
   4.7 Conclusion and Future Work .................................................................................. 69
5. High performance Computing ..................................................................................... 71
   5.1 Distributed Topic Modeling via Map Reduce .......................................................... 71
   5.2 Vectorization and Efficient Memory Usage .............................................................. 77
      5.2.1 Efficient Memory Usage ................................................................................ 78
      5.2.2 Vectorization .................................................................................................. 79
   5.3 Numerical Efficiency Strategies .............................................................................. 81
      5.3.1 Cholesky Decomposition for Large Matrix Inversion ...................................... 81
      5.3.2 Simplifying the Kronecker Product ................................................................ 81
6. Conclusion and Future Work ........................................................................................ 83
Bibliography ................................................................................................................. 84
Appendix ...................................................................................................................... 88
SIFT Implementation via Python .................................................................................... 88
Chapter 1

1. Introduction and Overview

The task of recovering the latent topics underlying a given corpus of documents has been in the forefront of active research in statistical machine learning for more than a decade, and continues to receive the dedicated contributions from many researchers from around the world. Since the introduction of Latent Dirichlet Allocation (LDA) [1] which is commonly referred as the topic models, various excellent extensions of topic models have been invented and shown to be successful in a wide variety of applications, ranging from information retrieval to image annotation and even predicting the future. Logistic Normal Correlated topic model (CTM) [2] is one of the extension model developed in 2007 in which a variational inference approach was taken to infer the model parameters. Subsequent works for Correlated Topic Model (CTM) concentrates on the scalability of the model and the exact inference methods using Markov Chain Monte Carlo (MCMC) Gibbs sampling was proposed by [3] and [4].

In this thesis, we developed an alternative correlated topic model (CTM) called Probit Normal Correlated Topic Model [5] to contribute to the topic modeling community. Our model performs very well and shows great potential in various applications such as latent topic uncovering and image captioning while achieving good scalability.
1.1 Motivation and Scope of Thesis

Correlated Topic model (CTM) is one of the extension models of Latent Dirichlet Allocation (LDA) which adds a correlation structure between latent topical spaces through a logistic transformation of multivariate Gaussian variables. The original Correlated Topic model (CTM) was proposed in 2007 and have gained great popularity since then due to its enriched and comprehensive representation in topical relationship. In Bayesian learning and traditional regression problems, Logistic and Probit models are very commonly employed and often compared extensively. However, in the context of Correlated topic model (CTM), Probit version of this model has not been studied mainly due to the high dependency issue between auxiliary variables and model parameters, and inefficient latent variable sampling especially in high dimensional space. Therefore, it motivates me to develop a viable Probit model in the context of topic modeling as it avoids unnecessary Independence of irrelevant alternatives (IIA) assumption [6] from the logistic models. In this thesis, I leverage a recent advancement in multinomial Probit model [7] and adapt it into our Probit Normal Correlated topic model, which successfully solve the high posterior conditional dependency and enable this model to be a viable alternative to the existing algorithms. This part of this thesis has already been published in the Open Journal of Statistics [5]. The next part of this thesis focuses on applying our proposed model to the state of art image annotation problem. In the context of image captioning or annotation, each image is paired with an article and the goal of this task is to annotate caption words for each unseen image [8]. The last part of this thesis focuses on methods of high performance computing employed in this thesis which is vital to this type of modeling as data normally comes with high dimension and consumes large
memory space. In addition, for applications like analyzing tweets, the algorithm needs to perform at least as fast as the update of new tweets, otherwise, it will lag further behind and the analysis would soon be outdated and become obsolete.
1.2 Thesis Organization

In Chapter 2, I introduce the background of topic models and several state-of-the-art text mining methodologies and terminologies as well as preprocessing techniques employed in this thesis. Chapter 3 details the Markov Chain Monte Carlo (MCMC) Gibbs Sampling Schemes for Probit Normal Correlated Topic Model and its implementation in topic recovery tasks, in addition, I also show the progression of the Markov Chain Monte Carlo (MCMC) Gibbs sampling schemes before and after the adaption of Diagonal Orthant Multinomial Probit Models [7] to demonstrate the high posterior dependency issue in the original Multinomial Probit Models. Chapter 4 describes the task of image annotation and the implementation of our model. It also shows the Markov Chain Monte Carlo (MCMC) Gibbs sampling for both Probit Normal Correlated Topic Model and the traditional Logistic Normal Correlated Topic Model. Chapter 5 describes various techniques employed in this thesis to make the inference task scalable with large datasets. Chapter 6 concludes the thesis and lays out several pointers to future work. The appendix includes the code for extracting visual features using SIFT
Chapter 2

2. Thesis Background

In this section, I give a short introduction to the background of topic modeling and text mining to facilitate the understanding for Chapter three and four.

The booming of the internet has overloaded the world with abundant information such as Wikipedia, Facebook, Twitter and etc. These data often appear in an unorganized fashion and a slew of preprocessing has to be done before feeding into algorithms such as Topic models or other semantic models. We could call this “preprocessing” as a typical Text Mining and Natural Language Processing (NLP) [9] task and it is actually one of the most crucial steps in probabilistic semantic models since most probabilistic models of interest in this thesis is sensitive to this preprocessing of raw text data.

Latent Dirichlet Allocation (LDA), represented as a probabilistic graphical model, was first introduced in 2003 to learn from a large number of documents and then represents each document as a mixture of “topics” in which each topic is a distribution over its corresponding words. One could view this algorithm as a dimensionality reduction technique for text data as it reduced the dimension of each document from a collection of words to a mixture of topics, improving information retrieval and text classification. In addition, LDA is often used as an entry point technique for various research topics such as sentiment analysis.
2.1 Text Mining and Natural Language Processing

As stated above, text data usually appears in an unorganized form, therefore, we need to extract the data and then process them in such a way to facilitate modeling of interest. In topic modeling, text is represented as document term matrix which will be introduced formally in the next section. Here I introduce a state of the art pipeline of Natural Language Processing [10].

- The correct format of character encoding (ASCII, Unicode and etc.) needs to be determined before processing text in. For example, the pound sign £ needs to be handled correctly before feeding into R programs while in Python, “\xa3” automatically replaced the pound sign £ if no encoding is used.

- The language of the input text data is determined since different language have different stop words and stemming methods.

- The type of the data is detected and then a suitable text scraping technique should be used to extract the text information. For example, some text files are stored as HTML hence “tm” and “XML” packages [11, 12] should be employed to scrape the data into R.

- Depending on the modeling of interest, words are processed either as token or n-gram. Tokenization represents each word as its own while n-gram put n words together as one word. For instance, sentence “This food is not good” can be tokenized into “this”, “food”, “is”, “not”, “good”. However for n-gram, we could put “not-good” together as one 2-gram word. In the traditional setting of topic model, words are usually represented in tokenized form as understanding the semantic meaning of each word is not necessary to infer the model parameters.
However in some applications such as sentiment analysis, n-grams are vital as treating “not-good” together instead of separately has completely different semantic meaning.

- Stop words like “and”, “is”, “are” should be removed to enhance the learning task of interest and one should be aware that the stop words list could vary from task to task.

- Stemming and lemmatization grouped same words of all forms together as a linguistically correct word. For example, “studies”, “studied”, “studying” are grouped together as “studi”. However, one may notice that in some context, this stemming needs to be changed as “studied” means past events while “studying” indicates current events.

- A Part-Of-Speech Tagger (POS Tagger) reads text and assigns parts of speech to each token, such as noun, verb, adjective which facilitate semantic learning in models like Hidden Markov Model.

- Lastly, similar words are tagged as synonyms while on the other hand same words with different meanings are tagged as homonyms. One such example is “bow”.

Fortunately, we could use any major statistical software such as R and Python to complete all the complicated tasks listed above for us. Public tools like Stanford NLP and MALLET are also available to process large number of text file.
Figure 1 Pipeline for Natural Language Processing ([10])
2.2 The Original Topic Model - Latent Dirichlet Allocation

2.2.1 Introduction with a Motivating Example

Latent Dirichlet Allocation, a generative hierarchical Bayesian model, models a collection of text corpora or documents to represent each document in the corpus as a mixture of “topics”, and each “topic” is in fact also a distribution of the words that have different probability to occur when this “topic” is active. Each “topic” contains a collection of words that are likely to co-occur with each other. Therefore, the key questions Latent Dirichlet Allocation tries to answer are the following:

- For each document, what are the topic proportions?
- For each topic, what is the probability for each word to occur?

I herein introduce a motivating example before laying out the mathematical detail of Latent Dirichlet Allocation. Suppose we have the following three sentences and we consider each sentence as a separate document for the purpose of illustration.

1. I eat vegetable and fruits.
2. Dogs and cats are so cute.
3. I don’t eat dogs.

By using Latent Dirichlet Allocation for these three documents and allowing only two topics to occur for each document, we would obtain the following results.

1. I eat vegetable and fruits.
2. Dogs and cats are so cute.
3. I don’t eat dogs.
• Topic 1: eat, vegetable, fruits

• Topic 2: Dogs, cats, cute

• Topic proportions of document 1: 100% Topic 1, 0% Topic 2

• Topic proportions of document 2: 0% Topic 1, 100% Topic 2

• Topic proportions of document 3: 50% Topic 1, 50% Topic 2

From this simple motivating example, we observe that the Latent Dirichlet Allocation projects each document from a collection of words to a mixture of topics. Imagine in the case of the entire Wikipedia dataset containing over 4.6 million documents and 811 million words, with Latent Dirichlet Allocation, each document is now represented as a vector of topic proportions, which drastically reduces the original dimension of the input space.

The original intention of Latent Dirichlet Allocation is to perform document classification, document retrieval and efficient recommender systems. In the age of big data, information retrieval becomes extremely challenging owning to the overflow of unstructured and noisy data. Thus, automatic and accurate classification would greatly improve the efficiency of information retrieval. For example, JSTOR, a non-profit organization for online journal archive, maintains a large database of printed journal by running optical recognition systems [2]. Due to the massive volume of the journals, it would not be attainable for staff at JSTOR to read all of the articles and organize them for researcher to search articles of interest. Hence an automatic algorithm of grouping similar documents is essential to maintain this type of large corpus. Fortunately with the help of Latent Dirichlet Allocation, we are able to classify the documents based on the topic proportions extracted for each document, which greatly improve the accuracy and
efficiency of organizing new documents. This algorithm can also be programmed as an online learning tool that will update model parameters every time after incorporating a new batch of documents, providing significant convenience for online journal Archive such as JSTOR and ARXIV. In addition, we could perform various similarity measures between documents or using collaborative filtering to automatically recommend users for additional articles that they might be interested in terms of the topics presented in each article. Furthermore, Latent Dirichlet Allocation naturally lends itself to various applications such as image annotation or image retrieval as images are essentially a collection of visual features. Hence, each image can be represented as a mixture of visual topics. Chapter four of this thesis will describe the pipeline of image annotation in great detail.

2.2.2 Model Input and Assumptions

Assumptions

Recall the motivating example from previous section, Latent Dirichlet Allocation assumes that a document is a collection of words or bag of words. The bag-of-word assumption also indicates that the ordering of the word does not matter, which is reasonable in the case of uncovering course semantic relationship between words and topics for document classification and retrieval since the joint distribution of the parameters is invariant to permutation. For more sophisticated applications as sentiment analysis, language generation, this assumption needs to be relaxed to accommodate and reflect the actual modeling situation. The second assumption of Latent Dirichlet Allocation is the ordering of the document does not matter, this assumption is also
reasonable for general purpose applications that does not involve learning topics from different time span. In the case of modeling changing topic overtime and predicting future topics also known as the dynamic topic modeling [13], the ordering of the document has to be taken into account as now a topic is an ordered sequence of words. A third assumption is that the number of topics is predetermined and fixed before implementing the model. This assumption can be overcome by cross validation via comparing performance measures for each different setting of topics. One extension of Latent Dirichlet Allocation such as Max Margin Topic models [14] allows different global and local topics to relax this assumption, which in turn delivers outstanding results. One example would be modeling Yelp comments. We could have “Food”, “Service”, and “Ambience” as the global topics. We could add another topic “Speed” for restaurant that offers delivery. In addition, [15] developed a model that completely eliminates the need for cross validating this parameter as the model itself will determine the optimal number of topics in the learning process. Since for industry size dataset, cross validating could be very time consuming and inefficient, this method offers great insight and speed for large corpus.

**Model Input**

As stated in the above section, one essential assumption is the bag-of-words in which each document is represented as a collection of words and the ordering of word does not influence the output of model parameters. Therefore, the first step is to tokenize each word for each document. Secondly, stop words and words with length less than three are removed as these words are normally considered as noise in the context of Nature
Language Processing. Thirdly, stemming and lemmatization is performed so that similar words are grouped together.

1. I eat vegetable and fruits. I, eat, vegetable, and, fruits eat, vegetable, fruit
2. Dogs and cats are so cute. Dogs, and, cats, are, so, cute dog, cat, cute
3. I don’t eat dogs. I, don’t, eat, dogs eat, dog

To represent each document as a collection of words, we introduce the state of the art document term matrix or term document matrix depending on the orientation of the matrix (document term matrix is the transpose of term document matrix. For the motivating example, the document term matrix is shown in the following table. Each entry of the document term matrix represents the frequency of the words that appear in its corresponding document. With the document term matrix in hand, we could proceed to the formal analysis of Latent Dirichlet Allocation.

<table>
<thead>
<tr>
<th></th>
<th>cat</th>
<th>cute</th>
<th>dog</th>
<th>eat</th>
<th>fruit</th>
<th>vegetable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document 1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Document 2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Document 3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### 2.2.3 Generative Process of Latent Dirichlet Allocation

Formally, we define the following terminology used throughout the entire thesis:

- Each document $d$ contains $N_d$ number of words, we use $V$ to denote the total number of words in the entire corpus. Using document term matrix we presented in Table 1, a document is represented by a vector of 0 and 1 where 1 denotes the presence of the corresponding word
A corpus is a collection of \( d \) number of documents, we use \( D \) to denote the total number of documents in the corpus.

Latent Dirichlet Allocation, a probabilistic graphical model, assumes each document \( d \) is created from the following generative process and the basic terminology used in the model is shown in Table 2:

1. Draw the topic proportions \( \theta_d \sim \text{Dirichlet}(\alpha) \):

2. For each word position \( n \in (1, \cdots, N_d) \):
   - Draw a topic \( z_{d,n} \sim \text{Multinomial}(\theta_d) \)
   - Draw a term \( w_n \sim \text{Multi}(\phi_k) \)

One may argue that a real document is not created by this process. However, with the bag-of-words assumption in which the ordering of the words does not matter, this generative process is reasonable as long as we are able to correctly infer the topic proportions \( \theta \) and the word-topic probability \( \phi_k \).

| \( V \)  | total number of words in the corpus |
| \( k \)  | topic index |
| \( K \)  | Total number topics |
| \( d \)  | document index |
| \( D \)  | Total number documents in the corpus |
| \( W_n \) | Term index in a document |
| \( \alpha \) | hyperparameter of Dirichlet for topic proportion \( \theta \) |
| \( \beta \) | hyperparameter of Dirichlet for word topic distribution \( \phi_k \) |
| \( \theta_d \) | topic proportion of document \( d \) |
| \( Z \)  | topic assignment vector, \( z_{d,n}^k = 1 \) means the word is drawn from topic \( k \) |
| \( z_{d,n} \) | topic assignment of word \( n \) in document \( d \) |
| \( \phi_k \) | word topic distribution of topic \( k \) |
The probabilistic graphical representation of the model is shown as follows in Figure 2.

![Figure 2 Probabilistic Graphical Model of LDA](image)

The probabilistic graphical model offers a clean-cut explanation of the generative process and eases the construction of the sampling schemes. The shaded node in the graph is the only subject observed in this model which is all the words presented in each document, while the empty node denotes the latent variables to be inferred from the model. The plates represented a repeated structure as the corpus contains $D$ number of documents, and each document contains $N$ words. In addition, each document is a mixture of $K$ topics and every topic is a distribution of words with probability $\phi_k$. Every node in the graph is linked with a directional arrow, indicating the dependency structure of the model. For instance, assuming we know the hyperparameter $\beta$, we could obtain the topic-word distribution for each topic $K$. Assuming also we know the topic hyperparameter $\alpha$, we could then draw $\theta \sim Dirichlet(\alpha)$ to obtain the topic proportions for each document. Then we could draw the topical assignment $z_n \sim Multinomial(\theta)$ for each word in the document. Finally, we could now sample for each word $w_n \sim Multi(\phi_{z_n})$ since we know both the topical assignment $z_n = k$ and the topic-word distribution $\phi_{z_n=k}$. Aside from
the dependency structure between variables, the probabilistic graphical model also encodes the preliminary assumptions of Latent Dirichlet Allocation introduced in section 2.2.2.

**Dirichlet Distribution**

Latent Dirichlet Allocation assumes a document is a mixture of topics. Therefore, the Dirichlet distribution is an ideal distribution to model the topic proportion \( \theta \). In addition, it is conjugate to the multinomial distribution which plays a vital role in the generative process of the model. A \( k \)-dimensional Dirichlet distributed random variable \( \theta \) lives in the space of a \((k - 1)\) simplex and has the probability density as follows:

\[
p(\theta|\alpha) = \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \alpha_i} \theta_1^{\alpha_1} \cdots \theta_k^{\alpha_k}, \quad \sum_{i=1}^{k} \theta_i = 1 \tag{2.1}
\]

Where the parameter \( \alpha \) is a \( k \)-dimensional vector with components \( \alpha_i > 0 \) and \( \Gamma(x) \) is the Gamma function. The parameter \( \alpha \) controls the shape of the distribution and in the context of Latent Dirichlet Allocation, it controls the sparsity of the topic proportions \( \theta \).

By convention, a symmetric Dirichlet distribution where all \( \alpha_i \) is equal is preferred since we have no prior knowledge of the topical distribution for the documents. In the Bayesian framework, this is referred to as a non-informative prior. From the examples in Figure 3, smaller \( \alpha \) value generates sparse topic distribution. In most practical cases, a smaller \( \alpha \) value should be chosen as the topic proportion should be calculated by the posterior distribution. However, in cases such as some supervised topic models, the \( \alpha \) parameter needs to be adjusted in terms of the prior knowledge of the topic proportion. Otherwise, a
small $\alpha$ value like 0.01 should be used. [16] suggests that $50/K$ is a good heuristic to set the initial $\alpha$.

Figure 3 Geometrical Representation of a Document with Three Topics ([10])
Parameter Estimation

From the generative process and probabilistic graphical model, once again the two problems Latent Dirichlet Allocation needs to solve are the following:

- For each document, what are the topic proportions?
- For each topic, what is the probability for each word to occur?

However, the only observed objects are the words from each document. Therefore the topic proportions $\theta$ and word topic distribution $\beta_k$ are the latent variables to be estimated.
Based on the probabilistic graphical model in Figure 2, the joint distribution of the model is shown as follows:

\[
p(w, z, \theta, \phi | \alpha, \beta) = p(\phi | \beta) p(\theta | \alpha) p(z | \theta) p(w | \phi_z)
\]  

(2.2)

Thus, the key aspect of Latent Dirichlet Allocation is to infer the following posterior distribution:

\[
p(z, \theta, \phi | \alpha, \beta) = \frac{p(z, \theta, \phi, w | \alpha, \beta)}{p(w | \alpha, \beta)}
\]  

(2.3)

The numerator could be decomposed into four components by the hierarchical structure of graphical model:

\[
p(z, \theta, \phi, w | \alpha, \beta) = p(\theta | \alpha) p(z | \theta) p(\phi | \beta) p(w | z, \phi)
\]  

(2.4)

Clearly, \(p(\theta | \alpha)\) and \(p(\phi | \beta)\) represents the probability density of the Dirichlet distributed topic proportion \(\theta\) and word topic distribution \(\phi\). \(p(z | \theta)\) is the probability of number of times topic \(k\) assigned to each word in the corpus. \(p(w | z, \phi)\) denotes the likelihood of observing a document given the probability \(\phi\) of each word in its corresponding topic \(z\).

Bringing all the components together we have:

\[
p(z, \theta, \phi, w | \alpha, \beta) = p(\theta | \alpha) p(z | \theta) p(\phi | \beta) p(w | z, \phi)
\]  

\[= \left( \prod_{d=1}^{D} \frac{\Gamma\left(\sum_{i=1}^{k} \alpha_i\right)}{\prod_{i=1}^{K} \alpha_i} \prod_{k=1}^{K} \theta_i^{a_i} \right) \left( \prod_{d=1}^{D} \prod_{k=1}^{K} \theta_{d,k}^{n_{d,k}} \right)
\]
\[
\left( \prod_{k=1}^{K} \frac{\Gamma(\sum_{i=1}^{k} \beta_i)}{\Pi_{i=1}^{k} \alpha_i} \prod_{k=1}^{K} \phi_{k,v}^{\beta_{k,v} - 1} \right) \left( \prod_{k=1}^{K} \prod_{v=1}^{V} \phi_{k,v}^{n_{k,v}} \right)
\]

(2.5)

Where \( n_{d,k} \) is the total number of words assigned to topic \( k \) in document \( d \) and \( n_{k,v} \) is the total number of times topic \( k \) is assigned to word \( v \) in the entire corpus.

Due to the conjugacy between Multinomial distribution and Dirichlet distribution, the joint distribution becomes:

\[
p(z, \theta, \phi, w|\alpha, \beta) = p(\theta|\alpha)p(z|\theta)p(\phi|\beta)p(w|z, \phi)
\]

\[
= \left( \prod_{d=1}^{D} \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\Pi_{i=1}^{k} \alpha_i} \prod_{k=1}^{K} \theta_{i}^{\alpha_{i}} \right) \left( \prod_{d=1}^{D} \prod_{k=1}^{K} \theta_{d,k}^{n_{d,k}} \right) \left( \prod_{k=1}^{K} \frac{\Gamma(\sum_{i=1}^{k} \beta_i)}{\Pi_{i=1}^{k} \beta_i} \prod_{k=1}^{K} \phi_{k,v}^{\beta_{k,v} - 1} \right) \left( \prod_{k=1}^{K} \prod_{v=1}^{V} \phi_{k,v}^{n_{k,v}} \right)
\]

(2.6)

I marginalize all the latent variables \( z, \theta, \phi \) to obtain the denominator of the posterior distribution of interest:

\[
p(w|\alpha, \beta)
\]

\[
= \int_{\phi} \int_{\theta} \sum_{z} \left( \prod_{d=1}^{D} \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\Pi_{i=1}^{k} \alpha_i} \prod_{k=1}^{K} \theta_{i}^{\alpha_{i} + n_{d,k} - 1} \right) \left( \prod_{k=1}^{K} \frac{\Gamma(\sum_{i=1}^{k} \beta_i)}{\Pi_{i=1}^{k} \beta_i} \prod_{k=1}^{K} \phi_{k,v}^{\beta_{k,v} + n_{k,v} - 1} \right) d\theta d\phi
\]

(2.7)

Unfortunately, this normalizing constant of the posterior is computationally intractable.

However, researchers have developed various distinct ways of solving this intractable posterior:
- Variational Inference with strict mean field assumption: This approach approximates the posterior by minimizing the Kullback–Leibler (KL) divergence between the proposal distribution and actual posterior [1,2]

- Expectation Propagation: This approach is an exact posterior inference using Belief Propagation (BP) [17]

- Collapsed Gibbs Sampling: This approach is an exact posterior inference, Markov Chain Monte Carlo sampling methods [16]

- Metropolis Hastings algorithm: This approach is an exact posterior inference, Markov Chain Monte Carlo (MCMC) sampling methods with a proposal distribution. This algorithm is the most efficient way to model industry size corpus as it has a constant order of complexity regardless of the topical space [18]

- Matrix Factorization based inference: This approach used Nonnegative Matrix Factorization (NMF) and requires only minimum assumption [19]

- Excess Correlation Methods: This approach used a spectral decomposition of low-order moments via two singular value decompositions (SVDs) [20]


In the scope of this thesis, I focus on the Gibbs sampling method which is one member of the Markov Chain Monte Carlo (MCMC) family. The idea of Gibbs sampling is intuitive and straightforward, yet delivering powerful performance for intractable posterior inference. Furthermore, Gibbs sampling offers exact solution for the target posterior distribution via its construction of stationary Markov chain. Specifically, the sampling should converge to the desired posterior after reaching the steady state of the Markov chain.
For instance, to sample $x$ from the joint distribution $p(x) = p(x_1, x_2, ..., x_n)$ where no closed form solution could be obtained, but a representation of full conditional distributions of $x$ is available through construction, one would perform the Gibbs Sampling for the intractable posterior distribution as follows:

1. Randomly choose the initial state of each $x$

2. For iteration $t$ from 1, ... $T$:
   1) $x_1^{t+1} \sim p(x_1 | x_2^t, x_3^t, ..., x_n^t)$
   2) $x_2^{t+1} \sim p(x_2 | x_1^{t+1}, x_3^t, ..., x_n^t)$
   3) $x_n^{t+1} \sim p(x_1 | x_1^{t+1}, x_2^{t+1}, ..., x_{n-1}^{t+1})$

3. The tuple $x_1^T, x_2^T, ..., x_n^T$ sampled from the conditional distribution is theoretically guaranteed to be samples from the joint distribution after the algorithm converges

4. The first moment of marginal distribution can be estimated by $E(x_i) = \frac{\sum_{j=1}^{M} x_i^{t+j}}{M}$ which is the expected value of $x_i$ when the Markov Chain reaches steady state after $t$ number of iterations

This sampling process repeated to a number of iterations until the distribution begins to converge to the true joint distribution. Despite that, Gibbs Sampling is theoretically guaranteed to converge. Diagnosing convergence requires setting the threshold for log-likelihood or particular performance measures that will assist the inspection of posteriors. By convention the initial $t$ number of iterations is called the burn-in period during which the Markov chain searches through the space of the posterior distribution. When the Markov chain reaches steady state, the expectation of
the parameter is calculated by averaging the values with certain lag to ensure the independence of the random variables. Heuristically, a lag of at least 3 or 5 is chosen to reduce first order correlation since each subsequent samples depends on the samples from the previous iteration. For example, from the sample path shown in Figure 5, we observe that the first 200 iterations is the burn-in phase during which the sampler explores the space of the distribution. By discarding the initial 200 burn-in samples, the parameter could be obtained by calculating the expected value with certain lag to ensure the independence of the random variables.

Figure 5 Sample Path of a Gibbs Sampler
However, in the case of complex posterior in which a large of number of parameters needs to be inferred, MCMC algorithms such as Gibbs Sampling tend to be computationally expensive due to the construction of full conditional distribution. Therefore, a compact Gibbs sampler could be obtained by integrating out some of the parameters. Formally, this is called Collapsed Gibbs Sampler. Fortunately, for the inference of the intractable posterior $p(z, \theta, \phi | \alpha, \beta) = \frac{p(z, \theta, \phi, w | \alpha, \beta)}{p(w | \alpha, \beta)}$, we can integrate out $\theta, \phi$ to construct the Collapsed Gibbs Sampler only for $z$, which drastically decreases the complexity of the sampling scheme. In addition, from the graphical model of Latent Dirichlet Allocation, knowing the topic assignment $z$ for each word in the corpus is a sufficient statistics to know the entire posterior distribution. As explained earlier, Gibbs Samplers progresses to the target distribution by the construction of full conditional distribution. In order to infer the posterior of topic assignment $z$, it is necessary to compute the following conditional distribution of $z$:

$$ p(z_i | z_{-i}, \alpha, \beta, W) $$

Where $z_{-i}$ means we take into account all the topic assignment except for the current topic assignment of $i$ since Gibbs sampler is constructed in terms of the full conditional distribution. Therefore, the Bayes rule implies the following:

$$ p(z_i | z_{-i}, \alpha, \beta, w) = \frac{p(z_i, z_{-i}, W | \alpha, \beta)}{p(z_{-i}, W | \alpha, \beta)} \propto p(z_i, z_{-i}, W | \alpha, \beta) = p(Z, W | \alpha, \beta) $$

$$ = \int \int p(Z, W, \theta, \phi | \alpha, \beta) d\theta d\phi $$

$$ = \int \int (\phi | \beta)p(\theta | \alpha)p(z | \theta)p(w | \phi_z) d\theta d\phi p $$
Both the integrals have the conjugacy between Multinomial and Dirichlet, which means we have:

\[
p(Z, W | \alpha, \beta) = \int p(z|\theta)p(\theta|\alpha)d\theta \int p(w|\phi)p(\phi|\beta)d\phi
\]

\[
= \left( \prod_{d=1}^{D} \frac{\Gamma \left( \sum_{i=1}^{K} \alpha_{i} \right)}{\prod_{i=1}^{K} \alpha_{i}} \prod_{k=1}^{K} \theta_{i}^{\alpha_{i}+n_{d,k}-1} \right) \left( \prod_{k=1}^{K} \frac{\Gamma \left( \sum_{i=1}^{K} \beta_{i} \right)}{\prod_{i=1}^{K} \beta_{i}} \prod_{k=1}^{K} \phi_{k,v}^{\beta_{k,v}+n_{k,v}-1} \right)
\]

\[
= \prod_{d} B \left( n_{d,\cdot} + \alpha \right) B \left( \alpha \right) \prod_{k} B \left( n_{k,\cdot} + \beta \right) B \left( \beta \right)
\]

Where \((\alpha) = \frac{\prod_{i=1}^{K} \alpha_{i}}{\Gamma \left( \sum_{i=1}^{K} \alpha_{i} \right)}\), \(n_{d,\cdot}\) is the total number of words in document \(d\) and \(n_{k,\cdot}\) is the total number of words assigned to topic \(k\). Similarly, we obtain

\[
p(z_{-i}, W | \alpha, \beta) = \prod_{d} B \left( n_{d,\cdot}^{-i} + \alpha \right) B \left( \alpha \right) \prod_{k} B \left( n_{k,\cdot}^{-i} + \beta \right) B \left( \beta \right)
\]

Where \(n_{d,\cdot}^{-i}\) is the total number of words in document \(d\) and \(n_{k,\cdot}^{-i}\) is the total number of words assigned to topic \(k\) without considering the current word \(i\). Therefore the full conditional for the topic assignment \(z\) is shown as below:

\[
p(z_{i}|z_{-i}, \alpha, \beta, w) = \frac{p(z_{i}, z_{-i}, W | \alpha, \beta)}{p(z_{-i}, W | \alpha, \beta)} = \frac{p(z | \alpha, \beta)p(w | z, \alpha, \beta)}{p(z^{-i} | \alpha, \beta)p(w^{-i} | z, \alpha, \beta)}
\]

\[
\propto \prod_{d} B \left( n_{d,\cdot} + \alpha \right) \prod_{k} B \left( n_{k,\cdot} + \beta \right) \prod_{k} B \left( n_{k,\cdot}^{-i} + \beta \right) \propto \frac{n_{d,k}^{-i} + \alpha_{k}}{n_{d,k}^{-i} + \beta_{w}} n_{k,w}^{-i} + \beta_{w}
\]
After the Gibbs Sampler converges, the posterior estimate of topic-word probability \( \phi_{k}^{wn} \) and topic proportion \( \theta_{d}^{k} \) can be estimated by the following formula respectively:

\[
\theta_{d}^{k} = \frac{n_{d,k} + \alpha_{k}}{n_{d,.} + K \alpha_{k}} \quad (2.13)
\]

\[
\phi_{k}^{wn} = \frac{n_{k,wn} + \beta_{wn}}{n_{k,.} + V \beta_{wn}} \quad (2.14)
\]
Model Evaluation Method

Model evaluation and selection is one of the most crucial steps in modeling. In the context of topic modeling, the total number of topics $K$ is determined by evaluating the performance measures of held-out likelihood of the test data. Held-out data or test set data is the documents that are only used in the test phase of the modeling. In addition, it is also needed to compare different classes of topic models. However, measuring the performance of topic models is also a research area by itself, and there is currently no rule of thumb solution exists as most methods have their pros and cons. So far, the following metrics are commonly used by researchers, and in the scope of this thesis, we focus on the log likelihood method:

- Log likelihood $[22]$ for Gibbs Sampling is defined by:

$$\log(p(w|z)) = k \log\left(\frac{\Gamma(V\delta)}{\Gamma(\delta)^V}\right) + \sum_{w=1}^{V} \left\{ \sum_{k=1}^{K} \log\left(\Gamma(n_{k,w} + \beta)\right) - \log\left(\Gamma(n_k + V\beta)\right) \right\}$$ (2.15)

- Perplexity $[1,22]$ for Gibbs Sampling is defined by:

$$Perplexity(D_{test}) = \left\{ \frac{-\sum_d \sum_w n_{w,d} \log(\sum_k \theta_d^k \beta_k^w)}{\sum_d \sum_w n_{w,d}} \right\}$$ (2.16)

Where $n_{w,d}$ is the number of times word $w$ occurs in document $d$

- Empirical likelihood $[17]$

- Left to right samplers $[23]$

Since all the performance measure is used for the held-out data, the topic proportion $\theta$ for unseen data needs to be inferred via the Gibbs Sampling once again where the topic-word distribution learned in the training phase is employed.
2.3 Correlated Topic Model

2.3.1 General Aspects

Correlated Topic Model (CTM) [2], an extension model of Latent Dirichlet Allocation, offers a more flexible distribution for topic proportions which introduces the correlation structure among the topics. The addition of topic correlation allows a more realistic model than the Latent Dirichlet Allocation in which topics are assumed to arise independently from each other. However, topics tend to co-occur with each other, hence extracting the topical correlation is very helpful in applications such as recommender systems as it recommends additional topics that one might not know about. After implementing the Correlated Topic Model, an exemplary correlation graph extracted is shown in Figure 6 as below.

![Figure 6 Correlation Structure between Topics ([2])](image-url)
The generative process of Correlated Topic Model assumes a document arises from the following steps:

1. Draw $\eta_d \sim \text{MVN}(\mu, \sigma)$ and transform $\eta_d$ into topic distribution $\theta_d$ where each element of $\theta$ is computed as follows:

$$\theta_k^d = \frac{e^{\eta_k^d}}{\sum_{j=1}^{K} e^{\eta_j^d}} \quad (2.17)$$

2. For each word position $n \in (1, \cdots, N_d)$
   a. Draw a topic assignment $z_{d,n} \sim \text{Mult}(\theta_d)$
   b. Draw a word $w_{d,n} \sim \text{Mult}(\phi_k)$

The corresponding probabilistic graphical model is shown in Figure 7

![Figure 7 Probabilistic Graphical Model for CTM](image)

Evidently, from both the probabilistic graphical model and the generative process, the only difference between Latent Dirichlet Distribution (LDA) and Correlated Topic Model (CTM) lies in the construction of topic proportion. Instead of sampling from a Dirichlet distribution with a prior $\alpha$, we obtain the topic proportion now via the logistic transformation of a multivariate Gaussian variable with parameters $\mu$ and $\Sigma$ in Correlated
Topic Model (CTM). Alternatively, the topic proportion is derived by the logistic-normal distribution [2]. The introduction of the multivariate Gaussian variable $\eta$ allows the representation of topical relationship though the covariance matrix $\Sigma$ which is later learned in the model inference step.

### 2.3.2 Parameter Estimation

The enriched representation of topic proportion not only allows the topical correlation but also relax the assumption from the Latent Dirichlet Distribution. However, by modifying the distribution from multinomial distribution to a logistic transform of a multivariate Gaussian variable, we lost the conjugacy between the Dirichlet and Multinomial Distribution. In Correlated Topic Model (CTM), the joint posterior distribution of interest after integrating out the word-topic distribution $\phi$ takes the following form:

$$
\begin{align*}
    p(\eta, Z | W) & \propto p(W | Z) \prod_{d=1}^{D} \left( \prod_{n=1}^{Z_{dn}} \theta_d^{Z_{dn}} \right) N(\eta_d | \mu, \Sigma) \\
    & \propto \prod_{k=1}^{K} \frac{\Gamma(C_k + \beta)}{\Gamma(\beta)} \prod_{d=1}^{D} \left( \prod_{n=1}^{Z_{dn}} e^{\eta_d^{n_k}} \right) N(\eta_d | \mu, \Sigma)
\end{align*}
$$

(2.18)

Where $C_k$ is the total number of words assigned to topic $k$.

Due to the non-conjugacy between the logistic-normal distribution and the multivariate Gaussian, data augmentation techniques are required to construct an appropriate Gibbs Sampler. Metropolis Hasting algorithm could be adapted to approximate the posterior via a proposal distribution. However, finding a suitable proposal distribution is very computational inefficient. [24] developed a Gibbs Sampling schemes by data
augmentation with uniform variables obtained from truncated Gaussian distribution, which may involve complex rejection criteria and in turn computationally expensive. [25] later developed another Gibbs Sampling by data augmentation through a sophisticated distribution called Polya-Gamma distribution [26]. They achieved significant improvement in terms of the scalability and are able to analyze the Wikipedia dataset with over 6 million documents in mere two hours using a 40-node computer cluster. The detailed Gibbs Sampling scheme for these methods is beyond the scope the thesis.

In the traditional setting of logistic regression, Probit version of the regression model is also compared extensively with the logistic model. Comparing to the logistic model, Probit model is more complex and becomes intractable in high dimension. However, the Probit model does not have the strict Independence of Irrelevant Alternatives (IIA) assumption. Independence of Irrelevant Alternatives (IIA) assumes that the introduction of new alternative will not affect the previous preference and mathematically, it assumes a independent covariance matrix between choices. For example, assume we have $A > B$, the introduction of choice $C$ will not change the preference between $A$ and $B$. Essentially, the IIA property assumes a diagonal covariance matrix between choices, which may be true in some cases. However, in the grand scheme of modeling, it is too restrictive for modeling complex application such as topic modeling. On the other hand, Probit model is not tractable in high dimension and in the context of topic modeling where sometimes thousands topics are chosen. Therefore since the introduction of Correlated Topic Model [2] in 2007, no Probit model has yet been proposed and all the subsequent works focused solely on the scalability of the model and MCMC sampling.
schems despite the inseperaable relationship between logistic and Probit model. In the next chapter, we develope a noval Probit normal correlated model by leveraging a recent advancement in the multinomial Probit model (MNP) [7]. Our Probit model [5] has natural conjugacy between Multivariate Gaussian variables and latent variables and does not require sophisticated sampling schemes for auxiliary variables. We also illustrate the reasons why the original multinomial Probit model prevents the Probit version of the correlated topic model being developed in the context of topic model.
Chapter 3

3. Probit Normal Correlated Topic Model

The content from this chapter has been published on the Open Journal of Statistics [5].

3.1 Introduction

From the previous chapter, we have seen that the logistic normal distribution has been adapted via the transformation of multivariate Gaussian variables to model the topical distribution of documents in the presence of correlations among topics. In this chapter, we propose a Probit normal alternative approach to modelling correlated topical structures. Our use of the Probit model in the context of topic discovery is novel, as many authors have so far concentrated solely of the logistic model partly due to the formidable inefficiency of the multinomial Probit model even in the case of very small topical spaces. We herein circumvent the inefficiency of multinomial Probit estimation by using an adaptation of the Diagonal Orthant Multinomial Probit in the topic models context, resulting in the ability of our topic model-ling scheme to handle corpuses with a large number of latent topics. An additional and very important benefit of our method lies in the fact that unlike with the logistic normal model whose non-conjugacy leads to the need for sophisticated sampling schemes, our approach exploits the natural conjugacy inherent in the auxiliary formulation of the Probit model to achieve greater simplicity. The application of our proposed scheme to a well-known Associated Press corpus and BBC corpus not only helps discover a large number of meaningful topics but also reveals the capturing of compellingly intuitive correlations among certain topics. Besides, our
proposed approach lends itself to even further scalability thanks to various existing high
performance algorithms and architectures capable of handling millions of documents.
Since the introduction of Latent Dirichlet Allocation (LDA) [1] and then the extension to
correlated topic models (CTM) [2], a series of excellent contributions have been made to
this exciting field, ranging from slight extension in the modelling structure to the
development of scalable topic modeling algorithms capable of handling extremely large
collections of documents, as well as selecting an optimal model among a collection of
competing models or using the output of topic modelling as entry points (inputs) to other
machine learning or data mining tasks such as image analysis and sentiment extraction,
just to name a few. As far as correlated topic models are concerned, virtually all the
contributors to the field have so far concentrated solely on the use of the logistic normal
topic model. The seminal paper on correlated topic model [2] adopts a variational
approximation approach to model fitting while subsequent authors like [24] propose a
Gibbs sampling scheme with data augmentation of uniform random variables. More
recently, [25] presented an exact and scalable Gibbs sampling algorithm with Polya-
Gamma distributed auxiliary variables which is a recent development of efficient
sampling of logistic model. Despite the inseparable relationship between logistic and
Probit model in statistical modelling, the Probit model has not yet been proposed,
probably due to its computational inefficiency for multiclass classification problem and
high posterior dependence between auxiliary variables and parameters. As for practical
application where topic models are commonly employed, having multiple topics is
extremely prevalent. In some cases, more than 1000 topics will be fitted to large datasets
such as Wikipedia and Pubmed data. Therefore, using MCMC Probit model in topic
modeling application will be impractical and inconceivable due to its computational inefficiency. Nonetheless, a recent work on diagonal orthant Probit model [7] substantially improved the sampling efficiency while maintaining the predictive performance, which motivated us to build an alternative correlated topic modeling with Probit normal topic distribution. On the other hand, Probit models inherently capture a better dependency structure between topics and co-occurrence of words within a topic as it doesn’t assume the IIA (independence of irrelevant alternatives) restriction of logistic models.

The rest of this chapter is organized as follows: in section 3.2, we present a conventional formulation of topic modelling along with our general notation and the correlated topic models extension. Section 3.3 introduces our adaptation of the diagonal orthant Probit model to topic discovery in the presence correlations among topics, along with the corresponding auxiliary variable sampling scheme for updating the Probit model parameters and the remainder of all the posterior distributions of the parameters of the model. Unlike with the logistic normal formulation where the non-conjugacy leads to the need for sophisticated sampling scheme, in this section we clearly reveal the simplicity of our proposed method resulting from the natural conjugacy inherent in the auxiliary formulation of the updating of the parameters. We also show compelling computational demonstrations of the efficiency of the diagonal orthant approach compared to the traditional multinomial Probit for on both the auxiliary variable sampling and the estimation of the topic distribution. Section 3.4 presents the performance of our proposed approach on the Associated Press data set, featuring the intuitively appealing topics discovered, along with the correlation structure among topics and the loglikelihood as a
function of topical space dimension. Section 3.5 deals with our conclusion, discussion and elements of our future work.

3.2 General Aspects of the Probit normal correlated topic model

As described in the previous chapter, the correlated topic model is an extension model of the Latent Dirichlet Allocation by introducing an Multivariate Gaussian variables that brings the correlation between topics. Specifically, the logistic-normal defines

$$\eta_d = (\eta^1_d, \eta^2_d, \cdots, \eta^K_d)$$

where the last element $\eta^K_d$ is typically set to zero for identifiability and assumes with $\eta_d \sim MVN(\mu, \Sigma)$ with

$$\theta^k_d = \Pr[z_{dn}^k = 1 | \eta_d] = f(\eta_d) = \frac{e^{\eta^k_d}}{\sum_{j=1}^K e^{\eta^j_d}}, \quad k = 1, 2, \cdots, K - 1 \quad \text{and} \quad \theta^K_d = \frac{1}{\sum_{j=1}^K e^{\eta^j_d}},$$

Also, $\forall n \in \{1, 2, \cdots, N_d\}$ and $z_{dn} \sim \text{Mult}(\theta_d)$, and $w_{dn} \sim \text{Mult}(\beta)$. With all these model components defined, the estimation task in correlated topic modelling from a Bayesian perspective can be summarized in the following posterior

$$p(\eta_d, Z | W, \mu, \Sigma) \propto p(W | Z) \prod_{d=1}^D \left\{ \prod_{n=1}^{N_d} p(z_{dn}) p(\eta_d | \mu, \Sigma) \right\}$$

$$= \prod_{k=1}^K \frac{\delta(C_k + \beta)}{\delta(\beta)} \prod_{d=1}^D \left\{ \left( \prod_{n=1}^{N_d} \theta^d_{zn} \right) N(\eta_d | \mu, \Sigma) \right\}, \quad (3.2)$$

where $\delta(\cdot)$ is defined using the Gamma function $\Gamma(\cdot)$ so for a $K$-dimension vector $u$. 
(3.3) provides the ingredients for estimating the parameter vectors $\eta_d$ that help capture the correlations among topics, and the matrix $Z$ that contains the topical assignments. For each document, $Z$ is a $K$ by $N_d$ matrix. Under the logistic normal model, sampling from the full posterior of $\eta_d$ derived from the joint posterior in (3.3) requires the use of sophisticated sampling schemes like the one used in [24, 25]. Although these authors managed to achieve great performances on large corpuses of documents, we thought it useful to contribute to correlated topic modelling by way of the multinomial Probit. Clearly, as indicated earlier, most authors concentrated on logistic-normal even despite non-conjugacy, and the lack of Probit topic modeling can be easily attributed to the inefficiency of the corresponding sampling scheme. In the most raw formulation of the multinomial Probit that intends to capture the full extent of all the correlations among the topics, the topic assignment probability is defined by (3).

$$\text{Pr}(z_{dn} = k) = \theta_d^k = \int \int \cdots \int \phi_k(u; \eta_d, R) du$$

(3.3)

The practical evaluation of (3.3) involves a complicated high dimensional integral which is typically computationally intractable when the number of categories is greater than 4. A relaxed version of (3.3), one that still captures more correlation than the logit and that is also very commonly used in practice, defines $\theta_d^k$ as

$$\delta(u) = \frac{\prod_{k=1}^K \Gamma(u_k)}{\Gamma\left(\sum_{k=1}^K u_k\right)}.$$
\[ \theta_d^k = \int_{-\infty}^{+\infty} \left\{ \prod_{j=1, j \neq k}^{K} \Phi(v + \eta_d^k - \eta_d^j) \right\} \phi(v) dv = E_{\nu(v)} \left\{ \prod_{j=1, j \neq k}^{K} \Phi(V + \eta_d^k - \eta_d^j) \right\}, \quad (3.4) \]

where \( \phi(v) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}v^2} \) is the standard normal density, and \( \Phi(v) = \int_{-\infty}^{v} \phi(u) du \) is the standard normal distribution function. Despite this relaxation, the multinomial Probit in this formulation still has major drawbacks namely: (a) Even when one is given the vector \( \eta_d \), the calculation of \( \theta_d^k \) remains computationally prohibitive even for moderate values of \( K \). In practice, one may consider using a monte carlo approximation to that integral in (3.4). However, such an approach in the context of a large corpus with many underlying latent topics renders the Probit formulation almost unusable. (b) As far as the estimation of \( \eta_d \) is concerned, a natural approach to sampling from the posterior of \( \eta_d \) in this context would be to use the Metropolis-Hasting updating scheme, since the full posterior in this case is not available. Unfortunately, the Metropolis in this case is excruciatingly slow with poor mixing rates and high sensitivity to the proposal distribution. However, a recent paper drastically improved the efficiency of the Metropolis Hasting by using a uniform proposal distribution [24] and reduced the sampling complexity to a constant value. In the case of Gibbs sampling, it turns out that an apparently appealing solution in this case could come from the auxiliary variable formulation as described in [27]. Unfortunately, even this promising formulation fails catastrophically for moderate values \( K \) as we will demonstrate in the subsequent section, due to the high dependency structure between auxiliary variables and parameters. Essentially, the need for Metropolis
is avoided by defining an auxiliary vector $Y_d$ of dimension $K$. For $n = 1, \cdots, N_d$, we consider the vector $z_{dn}$ containing the current topic allocation and we repeatedly sample $Y_{dn}$ from a $K$-dimensional multivariate Gaussian until the component of $Y_{dn}$ that corresponds to the non-zero index in $z_{dn}$ is the largest of all the components of $Y_{dn}$, ie.

$$
Y^*_{dn} = \max_{k=1, \cdots, K} \{ Y^k_{dn} \}.
$$

(3.5)

The condition in (3.5) typically fails to be fulfilled even when $K$ is moderately large. In fact, we demonstrate later that in some cases, it becomes impossible to find a vector $Y_{dn}$ satisfying that condition. Besides, the dependency of $Y_{dn}$ on the current value of $\eta_d$ further complicates the sampling scheme especially in the case of large topical space. In the next section, we remedy these inefficiencies by proposing and developing our adaptation of the diagonal orthant multinomial Probit.

**3.3 Diagonal Orthant Probit for Correlated Topic Models**

In a recent work, [7] developed the diagonal orthant Probit approach to multicategorical classification. Their approach circumvents the bottlenecks mentioned earlier and substantially improves the sampling efficiency while maintaining the predictive performance. The Diagonal Orthant Multinomial represents a categorical variable as a collection of binary variables. Assume $y$ to be an unordered categorical variable with $J$ classes and define $y = j \iff \{ y_j = 1 \} \cup \{ y_k = 0 \ \forall \ k \neq j \}$ where $\gamma$ is an independent binary variable. By the state-of-the-art latent formulation of binary variables, we have $z_j \sim f(\mu_j, \sigma)$ where $f$ is a probability density with location parameter $\mu$ and scale
parameter $\sigma$, and $\gamma_j = 1 \iff z_j > 0$. In the context of regression and classification, we have only one $\gamma_j = 1$. Therefore, $Z$ belongs to the set $\Omega = \bigcup_{j=1}^{J} \{ z \in \mathbb{R}^J : z_j > 0, z_k < 0, k \neq j \}$. Based on the Radon-Nikodym theorem, the joint distribution of $Z$ is:

$$f(z) = \frac{1(z \in \Omega) \prod_{j=1}^{J} f(z_j - \mu_j)}{\int_{\mathbb{R}^J} 1(z \in \Omega) \prod_{j=1}^{J} f(z_j - \mu_j) \, dz} \quad (3.6)$$

In fact the function of $Z$ induces the class probability on $y$. Hence the categorical probabilities could be calculated by:

$$P(y = j) = \frac{(1 - F(-\mu_j)) \prod_{k \neq j} F(-\mu_k)}{\sum_{j=1}^{J} (1 - F(-\mu_j)) \prod_{k \neq j} F(-\mu_k)} \quad (3.7)$$

If $f$ is the univariate normal probability density function, the result from (3.7) becomes the Diagonal Orthant Probit Model (DO-Probit) and the class probability for each category can now be calculated as:

$$\theta^j_d = \frac{(1 - \Phi(-\eta^k_d)) \prod_{j \neq k} \Phi(-\eta^j_d)}{\sum_{\ell=1}^{K} (1 - \Phi(-\eta^\ell_d)) \prod_{j \neq \ell} \Phi(-\eta^\ell_d)} \quad (3.8)$$

Comparing the formulation of (3.4) in which a large number of standard normal variables need to sampled, the DO-Probit approach drastically decreases the computational complexity of the calculation of the class probability. Furthermore, thanks to the binary partitioning of space of $Z$, the sampling of the auxiliary variable in the context of implementing MCMC also becomes a binary classification problem. Therefore, the high dependency issue resulted from (3.3) can be simplified by sampling $Y_j \sim N_+(\mu_j, 1)$ and $Y_k \sim N_-(\mu_k, 1)$ for $k \neq j$, where $N_+(\mu_j, 1)$ means we sample the variable from the truncated normal distribution above 0 while $N_-(\mu_k, 1)$ means we
sample the variable from the truncated normal distribution below 0. This formulation entirely avoids the computationally expensive rejection sampling and high posterior dependency between the auxiliary variable and model parameters. Essentially, the diagonal orthant Probit approach successfully makes the most of the benefits of binary classification, thereby substantially reducing the high dependency that made the condition (3.5) computationally unattainable. Indeed, with the diagonal orthant multinomial model, we achieved three main benefits

- A more tractable and easily computable definition of topic distribution \( \theta^k_d = \Pr(z_{dn} = k | \eta_d) \)
- A clear and straightforward and adaptable auxiliary variable sampling scheme
- The capacity to handle a very large number of topics due to the efficiency and low dependency.

Under the diagonal orthant Probit model, we have

\[
\theta^k_d = \frac{(1 - \Phi(-\eta^k_d)) \prod_{j \neq k} \Phi(-\eta^j_d)}{\sum_{\ell=1}^{K} (1 - \Phi(-\eta^\ell_d)) \prod_{j \neq \ell} \Phi(-\eta^j_d)}. \tag{3.9}
\]

Where \( \Phi(\cdot) \) represents the cumulative distribution of the standard normal, the generative process of our Probit normal topic models is essentially identical to logistic topic models except that the topic distribution for each document now is obtained by a Probit transformation of a multivariate Gaussian variable (3.6). As such, the generating process of a document of length \( N_d \) is as follows:
1. Draw $\eta \sim MVN(\mu, \Sigma)$ and transform $\eta_d$ into topic distribution $\theta_d$

where each element of $\theta$ is computed as follows:

$$
\theta_d^k = \frac{(1 - \Phi(-\eta_d^k)) \prod_{j \neq k} \Phi(-\eta_d^j)}{\sum_{i=1}^{K} (1 - \Phi(-\eta_d^i)) \prod_{j \neq i} \Phi(-\eta_d^j)}.
$$

2. For each word position $n \in (1, \cdots, N_d)$

   c. Draw a topic assignment $Z_n \sim Mult(\theta_d)$

   d. Draw a word $W_n \sim Mult(\phi^n)$

Where $\Phi(\cdot)$ represents the cumulative distribution of the standard normal. We specify a Gaussian prior for $\eta_d$, namely $(\eta_d \mid \cdots) \sim N_K(\mu, \Sigma)$. Throughout this chapter, we’ll use $\phi_K(\cdot)$ to denote the $K$-dimensional multivariate Gaussian density function,

$$
\phi_K(\eta_d; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^K \mid \Sigma \mid}} \exp\left\{ -\frac{1}{2} (\eta_d - \mu)^T \Sigma^{-1} (\eta_d - \mu) \right\}.
$$

To complete the Bayesian analysis of our Probit normal topic model, we need to sample from the joint posterior

$$
p(\eta_d, Z_d \mid W, \mu, \Sigma) \propto p(\eta_d \mid \mu, \Sigma) p(Z_d \mid \eta_d) p(W \mid Z_d).
$$

(3.10)

As noted earlier, the second benefit of the diagonal orthant Probit model lies in its clear, simple, straightforward yet powerful auxiliary variable sampling scheme. We take advantage of that diagonal orthant property when dealing with the full posterior for $\eta_d$ given by

$$
p(\eta_d \mid W, Z_d, \mu, \Sigma) \propto p(\eta_d \mid \mu, \Sigma) p(Z_d \mid \eta_d).
$$

(3.11)
While sampling directly from (3.9) is impractical, defining a collection of auxiliary variables $\mathbf{Y}_d$ allows a scheme that samples from the joint posterior

$$p(\eta_d, \mathbf{Z}_d, \mathbf{Y}_d \mid \mathbf{W}, \mu, \Sigma)$$

using the following:

For each document $d$, the matrix $\mathbf{Y}_d$ contains all the values of the auxiliary variables,

$$\mathbf{Y}_d = \begin{bmatrix}
Y_{d1}^1 & Y_{d1}^2 & \ldots & Y_{d1}^K \\
Y_{d2}^1 & Y_{d2}^2 & \ldots & Y_{d2}^K \\
\vdots & \vdots & \ddots & \vdots \\
Y_{d,N_d-1}^1 & Y_{d,N_d-1}^2 & \ldots & Y_{d,N_d-1}^K \\
Y_{d,N_d}^1 & Y_{d,N_d}^2 & \ldots & Y_{d,N_d}^K \\
\end{bmatrix}$$

Each row $Y_{dn} = (Y_{dn1}^1, \ldots, Y_{dnK}^k, \ldots, Y_{dnK}^K)$ of $\mathbf{Y}_d$ has $K$ components, and the diagonal orthant updates them readily using the following straightforward sampling scheme: Let $k$ be the current topic allocation for the $n$th word.

- For the component of $Y_{dn}$ whose index corresponds to the label of current topic assignment of word $n$ sample from a truncated normal distribution with variance 1 restricted to positive outcomes

  $$ (Y_{dn}^k \mid \eta_{dn}^k) \sim N_{+}(\eta_{dn}^k, 1) \quad z_{dn}^k = 1 $$

- For all components of $Y_{dn}$ whose indices do correspond to the label of current topic assignment of word $n$ sample from a truncated normal distribution with variance 1 restricted to negative outcomes

  $$ (Y_{dn}^l \mid \eta_{dn}^l) \sim N_{-}(\eta_{dn}^l, 1) \quad z_{dn}^l \neq 1 $$
Once the matrix $Y_d$ is obtained, the sampling scheme updates the parameter vector $\eta_d$ by conveniently drawing

$$(\eta_d | Y_d, A, \mu, \Sigma) \sim MVN(\mu_{\eta_d}, \Sigma_{\eta_d}) ,$$

where

$$\mu_{\eta_d} = \Sigma_{\eta_d} \left( \Sigma^{-1} \mu + X_d^T A^{-1} vec(Y_d) \right) \quad \text{and} \quad \Sigma_{\eta_d} = (\Sigma^{-1} + X_d^T A^{-1} X_d)^{-1} .$$

with $X_d = 1_{N_d} \otimes I_K$ and $vec(Y_d)$ representing the row-wise vectorization of the matrix $Y_d$. Adopting the fully Bayesian treatment of our Probit normal correlated topic model, we add an extra layer to the hierarchy in order to capture the variation in the mean vector and the variance-covariance matrix of the parameter vector $\eta_d$. Taking advantage of conjugacy, we specify a normal-Inverse-Wishart prior for $(\mu, \Sigma)$, namely,

$$p(\mu, \Sigma) = NIW(\mu_0, \kappa_0, \Psi_0, \nu_0) ,$$

meaning that $\Sigma | \nu_0, \Psi_0 \sim IW(\Psi_0, \nu_0)$ and $(\mu | \nu_0, \kappa_0, \Sigma) \sim MVN(\mu_0, \Sigma / \kappa_0)$ . The corresponding posterior is normal-inverse-Wishart, so that we can write

$$p(\mu, \Sigma | W, Z, \eta) = NIW(\mu', \kappa', \Psi', \nu') ,$$

where $\kappa' = \kappa_0 + D , \nu' = \nu_0 + D , \mu' = \frac{D}{D + \kappa_0} \overline{\eta} + \frac{\kappa_0}{D + \kappa_0} \mu_0 ,$ and

$$\Psi' = \Psi_0 + Q + \frac{\kappa_0}{\kappa_0 + D} (\overline{\eta} - \mu_0)(\overline{\eta} - \mu_0)^T ,$$

where

$$Q = \sum_{d=1}^{D} (\eta_d - \overline{\eta})(\eta_d - \overline{\eta})^T .$$
As far as sampling from the full posterior distribution of $Z_{dn}$ is concerned, we use the expression

$$p(z^k_{dn} = 1 \mid Z_{-n}, w_{dn}, W_{ndn}) \propto p(w_{dn} \mid z^k_{dn} = 1, W_{ndn}, Z_{-n}) \alpha^k_d \propto \frac{C^k_{dn} + \beta_{dn}}{\sum_{j=1}^{V} C^j_{dn} + \sum_{j=1}^{J} \beta_j} \alpha^k_d.$$ 

where the use of $C_{.-n}$ is used to indicate that the $n$th word is not included in the topic or document under consideration.

### 3.4 Computational Results on the Associated Press data

In this section, we used the WEell-Associated Press data set from [22] in R to uncover the word topic distribution, the correlation structure between various topics as well as selecting optimal models. The Associated Press corpus consists of 2244 documents and 10473 words. After preprocessing the corpus by picking frequent and common terms, we reduced the size of the vocabulary from 10473 to 2643 for efficient sampling.

In our first experimentation, we built a correlated topic modelling structure based on the traditional multinomial Probit and then tested the computational speed for key sampling tasks. The high posterior dependency structure between auxiliary variables and parameters make multinormal Probit essentially unscalable for situations where it is impossible for the sampler to yield a random variate of the auxiliary variable corresponding the current topic allocation label that is also the maximum (3.5). For a random initialization of topic assignment, the sampling of auxiliary variable cannot even complete one single iteration. In the case of good initialization of topical prior $\eta_d$ which leads to smooth sampling of auxiliary variables, the computational efficiency is still undesirable and we observed that for larger topical space such as $K=40$, the auxiliary
variable stumbled again after some amount of iterations, indicating even good
initialization will not ease the troublesome dependency relationship between the auxiliary
variables and parameters in larger topical space. Unlike with the traditional Probit model
for which the computation of $\theta_d^j$ is virtually impractical for large $K$, the diagonal
orthant approach makes this computation substantially faster ever for large $K$. The
comparison of the computational speed of two essential sampling tasks between the
multinomial Probit model (MNP) and Digonal Orthant Probit model (DO-Probit) are
shown in the next page in Table 3.
Table 3 All the numbers in this table represent the processing time (in seconds), and are computed in R on PC using a parallel algorithm acting on 4 CPU cores. NA here represents situations where it is impossible for the sampler to yield a random variate of the auxiliary variable corresponding to the current topic allocation label that is also the maximum. All the measurement here is based on non-vectorized formulation for fair comparison, in the case of fully vectorization, the speed for DO Probit takes almost constant time and is less than 0.01 for k<100

<table>
<thead>
<tr>
<th>Sampling Task (K=10)</th>
<th>MNP</th>
<th>DO Probit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic Distribution $\theta$</td>
<td>18.3</td>
<td>0.06</td>
</tr>
<tr>
<td>Auxiliary variable $Y_d$</td>
<td>(108 to NA)</td>
<td>3.09</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sampling Task (K=20)</th>
<th>MNP</th>
<th>DO Probit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic Distribution $\theta$</td>
<td>63</td>
<td>0.13</td>
</tr>
<tr>
<td>Auxiliary variable $Y_d$</td>
<td>(334 to NA)</td>
<td>3.39</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sampling Task (K=30)</th>
<th>MNP</th>
<th>DO Probit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic Distribution $\theta$</td>
<td>123</td>
<td>0.21</td>
</tr>
<tr>
<td>Auxiliary variable $Y_d$</td>
<td>(528 to NA)</td>
<td>3.49</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sampling Task (K=40)</th>
<th>MNP</th>
<th>DO Probit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic Distribution $\theta$</td>
<td>211.49</td>
<td>0.33</td>
</tr>
<tr>
<td>Auxiliary variable $Y_d$</td>
<td>(1785 to NA)</td>
<td>3.79</td>
</tr>
</tbody>
</table>

In addition to the drastic improvement of the overall sampling efficiency, we noticed that the computational complexity for sampling the auxiliary variable and topic distribution is close to $O(1)$ and $O(K)$ respectively, suggesting that Probit normal topic model now becomes an attainable and feasible tool of the traditional correlated topic model.

Central to topic modelling is the need to determine for a given corpus the optimal number of latent topics. As it is the case for most latent variable models, this task can be formidable at times, and there is no consensus among machine learning researchers as to
which of the existing methods is the best. Figure 8 shows the loglikelihood as a function of the number of topics discovered in the model. Apart from the loglikelihood, many other techniques are commonly used such as perplexity, harmonic mean method and so on.

As we see, the optimal number of topics in this case is 30. In Table 4, we show a subset of the 30 topics uncovered where each topic is represented by the 10 most frequent words. It can be seen that our Probit normal topic model is able to capture the co-occurrence of words within topics successfully. In Figure 9, we also show the correlation structure between various topics which is the essential purpose of employing the correlated topic model. Evidently, the correlation captured intuitively reflect the natural relationship between similar topics.

![Figure 8 Loglikelihood as a Function of Topics](image)
### Table 4 Representation of Topics Discovered

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>court</td>
<td>company</td>
<td>bush</td>
<td>students</td>
<td>tax</td>
<td>fire</td>
<td>water</td>
<td>plane</td>
<td>student</td>
<td>illion</td>
<td>senate</td>
<td>schools</td>
<td>budget</td>
<td>bill</td>
</tr>
<tr>
<td>trial</td>
<td>billion</td>
<td>senate</td>
<td>schools</td>
<td>budget</td>
<td>billion</td>
<td>northern</td>
<td>airlines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>judge</td>
<td>inc</td>
<td>vote</td>
<td>meese</td>
<td>meese</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>prison</td>
<td>corp</td>
<td>dukakis</td>
<td>student</td>
<td>bill</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>convicted</td>
<td>percent</td>
<td>percent</td>
<td>schools</td>
<td>percent</td>
<td>southern</td>
<td>pilots</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>jury</td>
<td>stock</td>
<td>bill</td>
<td>teachers</td>
<td>senate</td>
<td>inches</td>
<td>aircraft</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>drug</td>
<td>workers</td>
<td>kennedy</td>
<td>board</td>
<td>income</td>
<td></td>
<td>fair</td>
<td>planes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>guilty</td>
<td>contract</td>
<td>sales</td>
<td>education</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>fbi</td>
<td>companies</td>
<td>bentsen</td>
<td>teacher</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sentence</td>
<td>offer</td>
<td>ticket</td>
<td>tax</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>percent</td>
<td>space</td>
<td>military</td>
<td>soviet</td>
<td>aid</td>
<td>police</td>
<td>arrested</td>
<td>dollar</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stock</td>
<td>shuttle</td>
<td>china</td>
<td>gorbachev</td>
<td>rebels</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>index</td>
<td>soviet</td>
<td>chinese</td>
<td></td>
<td>contras</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>billion</td>
<td>nasa</td>
<td>soldiers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>prices</td>
<td>launch</td>
<td>troops</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rose</td>
<td>mission</td>
<td>saudi</td>
<td>summit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>rose</td>
<td>mission</td>
<td>saudi</td>
<td>summit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stocks</td>
<td>earth</td>
<td>trade</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>average</td>
<td>north</td>
<td>rebels</td>
<td>treaty</td>
<td>ortega</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>points</td>
<td>korean</td>
<td>hong</td>
<td>europe</td>
<td>sandinistas</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>shares</td>
<td>south</td>
<td>army</td>
<td>germany</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>


Figure 9 Correlation Structures of Topics
3.5 Conclusion and Discussion

In the context of topic modelling where many other researchers seem to have avoided it, by adapting the diagonal orthant Probit model, we proposed a Probit alternative to the logit approach to the topic modeling. Compared to the multinomial Probit model we constructed, our topic discovery scheme using diagonal orthant Probit model enjoyed several desirable properties; First, we gained the efficiency in computing the topic distribution; Second, we achieved a clear and very straightforward and adaptable auxiliary variable sampling scheme that substantially reduced the strength of the dependence structure between auxiliary variables and model parameters, responsible for absorbing state in the Markov chain; Thirdly, as a consequence of good mixing, our approach made the Probit model a viable and competitive alternatives to its logistic counterpart. In addition to all these benefits, our proposed method offers a straightforward and inherent conjugacy, which helps avoid those complicated sampling schemes employed in the logistics normal Probit model.

In the Associated Press example explored in the previous section, not only does our method produce a better likelihood than the logistic normal topic model with variational EM, but also discovers meaningful topics along with underlying correlation structure between topics. Overall, the method we developed in this chapter offers another feasible alternatives in the context of correlated topic model.

Based on the promising results we have seen in this chapter, the Probit normal topic model opens the door for various future works. For instance, [28] proposed a multi-field
correlated topic model by relaxing the assumption of using common set of topics globally among all documents, which can also be applied to the Probit model to enrich the comprehensiveness of structural relationships between topics. Another potential direction would be to enhance the scalability of the model. Currently we used a simple distributed algorithm proposed by [29] and [30] for efficient Gibbs sampling. The architecture for topic models presented by [31] can be further utilized to reduce the computational complexity substantially while delivering comparable performance. Furthermore, a novel sampling method involving the Gibbs Max-Margin Topic [14] will further improve the computational efficiency.
Chapter 4

4. Image Annotation via the Probit Normal Correlated Topic Model

4.1 Introduction and Related Works

Traditional methods for information retrieval centered on text-based data such as documents, blogs, and tweets. However, the booming of the internet gave birth to another major type of data, images. To accommodate this growing trend, developing an efficient and automatic algorithm to retrieve images has become an active research area in computer vision and since then many appealing software and algorithms have been created in the past decade. For example, Google allows users to search images by both text and image query and the result is remarkably accurate. A robust deep learning algorithm developed by Baidu research is able to recognize an image even if the image is presented in noisy environment. To be able to retrieve an image, an appropriate representation of images needs to be developed to allow the interaction between a query and an image. Hence one such way is to annotate an image with some descriptive words, facilitating image retrieval through queries based search. Common way of image annotation includes metric learning, similarity measurement between images, probabilistic topic models and object reorganization/classification. In addition, a recent algorithm using convolutional neural networks (CNN) developed by Stanford Computer Vision Lab [32] describes an image using a sentence with semantic structure. In this chapter, we focus on the image annotation using probabilistic topic models and specifically, the Probit Normal Correlated Topic model we developed in the previous chapter.
Image annotation through probabilistic topic models is originally proposed by [33] in 2003 and many subsequent works have successfully demonstrated the impact in the literature of image retrieval and annotation. [34] extended the topic model to be able to classify and annotate image simultaneously. [35] developed a hierarchical probabilistic model with background distribution for biomedical images. [36] created a complex probabilistic topic model called the probabilistic topic connection model which incorporates various layers to represent the image-topic relationship. [37] developed correlated topic model based approach to improve the image annotation performance from the enriched representation of topical relationship. Later [38] extends the image annotation using correlated topic model to complex action recognition in motion pictures. Thanks to the flexible representation of probabilistic graphical model, all of the aforementioned extension works of topic models have showed great promises in the context of complex computer vision problems. Although logistic normal correlated topic models have been already implemented in the context of image annotation and have achieved superior results than its predecessor models, no exact inference such as Gibbs Sampling has yet to be proposed for correlated topic model. Therefore, in this chapter, we develop an exact inference method using MCMC Gibbs sampling for our proposed Probit Normal Correlated Topic Model. In addition, our sampling schemes naturally lend itself for logistical normal correlated topic model as the only difference lies in the generation of the prior parameter of topic proportion.

**4.2 Model Input and Data Description**

The model input for image annotation is a tuple of image and words. Specifically, each image comes with its corresponding descriptive words. In this chapter, we used the BBC
dataset from [39] in which each news article is paired with an image describing the content of the news. The ultimate goal of image annotation is to find an appropriate word from the corpus to describe the content of an unseen image. An example of the BBC dataset is shown as follows:

![Image](image_url)

Aviation history will be made in London as the world’s biggest passenger jet touches down at the world’s biggest international airport.

The 555-seat Airbus A380 will fly from Berlin to Heathrow for its first UK visit, after taking a minor detour.

The 240ft long plane will dip its wings as it flies over the Airbus sites that designed and made them, at Filton in North Wales and Filton, near Bristol.

It will test facilities at Heathrow before flying out on Friday.

**Aviation milestone**

Heathrow operator BAA is spending £450m so that it is ready to handle the A380 when it starts making commercial flights later this year.

The aircraft will be met at the airport’s new £105m pier 6 at Heathrow’s Terminal 3, and money has also been spent resurfacing runways, upgrading lighting and building new taxiways.

Developed by the European Airbus consortium for about £6bn, the A380 has been heralded as a major milestone in aviation history.

The jet maker has taken 159 firm orders for the plane from 16 carriers, including Singapore Airlines, which will make the inaugural commercial flight on its route between Sydney, Singapore and London later this year.

A spokesman for the world’s airport operators, told BBC Radio Four Live that the A380 was more environmentally friendly than older aircraft.

“It’s an efficient and clean and environmentally friendly aircraft, and it has lower fuel consumption per seat, and produces less noise and emissions than older aircraft, and for airports it should increase their ability to handle passengers,” said David Comper of the Airports Council International.

*Figure 10 Example from BBC dataset*

In the BBC dataset, there are 3361 images paired with 3361 articles, each article contains an average of 134 words. The vocabulary size is around 10,000 after removing stop
words and infrequent terms and the total number of words is 0.5 million. This data set is filled with noise and is considered difficult and challenging for image annotation. We also noticed that some of the images contain only the face of a person while the corresponding article describes a murder charge or other incidents that could be not be inferred from inspecting mere the images. An example of this is shown in the following Figure 11.

![Image](image_url)

The alleged head of the Sicilian mafia, Bernardo Provenzano, has made his first court appearance since his arrest last month after 40 years on the run.

Mr Provenzano, 73, appeared via a video-link from his jail in a trial concerning Mafia murders committed in Italy in the 1980s.

He is being held in isolation at a high security jail in Termini, central Italy.

He was arrested last month at a small farmhouse near his home town of Corleone in Sicily.

Police had followed fresh laundry sent by his wife to the farmhouse.

**Convicted**

Mr Provenzano was shown on screen in the court alongside the man accused of being his predecessor as Mafia boss, Toto “The Beast” Riina.

Before the session was adjourned, Mr Provenzano was shown via the video telling his lawyer that he was being treated well in the Termini prison.

The prison is denying him access to television, radio or newspapers.

As part of the tough prison regime reserved for Mafia convicts, he is under constant video surveillance and is only allowed contact with his lawyer.

Before his capture he had already been convicted in absentia of more than a dozen murders, and 10 more arrest warrants are being re-examined.

Dozens of letters and documents discovered in the farmhouse where he was hiding are being deciphered.

Police say important sections of the documents are in code or use a series of numbers to disguise names.

*Figure 11 Example of Noisy Input*
4.3 Scale Invariant Feature Transform (SIFT)

From section 2.2.2, we know the general assumption of topic model in which the model input is a discrete bag-of-words representation. However, in the case of image annotation, how do we represent each image as a collection of visual words? One solution is using the Scale invariant feature transform (SIFT) algorithm [40, 41] as it represent an image in the form of Bag-of-Features (BOF). In this section, I give a short introduction of Scale invariant feature transform (SIFT) and focus on high level aspects of this algorithm for the coherence of illustrating image annotation using our proposed model.

Formally, a Laplacian of Gaussian (LOG) [42, 43, and 44] is detected in the scale space in which the LOG represents a blob detector to detect blob features in various sizes. However, the Laplacian of Gaussian (LOG) detector is computationally expensive, therefore, [40] proposed the Difference of Gaussians (DOG) which is a fast and efficient approximation of LOG obtained by the DOG blurring of an image. By repeating the DOG blurring for different octaves of an image in Gaussian Pyramid as follows:
With Difference of Gaussian (DOG) obtained, local extrema over scale space is searched and once a local extremum is found, it serves as a potential keypoints, indicating it is one the best representation of the image in that scale.
Next, all the potential key points are refined by Taylor series expansion in the scale space to gain more accurate local extrema. The Difference of Gaussian (DOG) is sensitive in detecting edges in an image, therefore a Harris corner detector is used to remove edges. Then an orientation is calculated and assigned to each keypoint to allow the invariance to the rotation of the image, and then similar optimization methods are used to allow SIFT to have other properties. Finally, a keypoint descriptor is created to extract the corresponding image component of each keypoint. The original author [41] gives a further detailed and intuitive explanation of this process.

Generally, scale invariant feature transform (SIFT) extracts robust and distinctive features called keypoints from an image and in addition, these keypoints are invariant to image scale, rotation, distortion, change in 3D view point and illumination and the addition of noise. Therefore, similar features could be matched accurately over a large feature database extracted from many images. Furthermore, by clustering similar image features together, we are able to obtain a visual vocabulary and represent each image as bag of visual features.

The following flowchart vividly describes this procedure:

- Extract SIFT keypoints descriptors from a large number of images
- Cluster all the keypoints extracted using K-Means algorithms
- Obtain the visual vocabulary and represent each image as bins of an histogram
In this thesis, we used the open source computer vision tools OpenCV via Python to extract the SIFT features for each image. OpenCV is one of the most popular open source computer vision platforms among researchers and practitioners. In addition to OpenCV, public tools such as VLFeat and OpenIMAJ also provides the implementation of the SIFT algorithms as well as other major computer vision implementations.

### 4.4 Generative Process

In the traditional setting of topic modeling, only a collection of words represented as document term matrix is given as the model input. On the other hand, in the task of image annotation, a collection of visual features is given along with its corresponding
documents. Adapting from similar representations, both the bag of visual features and words are represented as the document term matrix.

We formally define the generative process of the general correlated topic model as follows:

1. Draw the topic proportions $\theta \sim f(\eta)$ where $\eta \sim MVN(\mu, \Sigma)$

$$\theta_{d}^{k} = f(\eta) = \frac{e^{\eta_{d}^{k}}}{\Sigma_{j}^{k} e^{\eta_{j}^{k}}} \text{ for logistic transformation}$$

$$\theta_{d}^{k} = f(\eta) = \frac{(1-\Phi(-\eta_{d}^{k})) \prod_{j \neq k}^{K} \Phi(-\eta_{d}^{j})}{\Sigma_{j=1}^{K}(1-\Phi(-\eta_{d}^{k})) \prod_{j \neq k}^{K} \Phi(-\eta_{d}^{j})} \text{ for Probit transformation}$$

2. For each visual feature $v_{m} \in \{v_{1}, \cdots, v_{M}\}$ :
   - Draw a topic $z_{m} \sim \text{Multinomial}(\theta)$
   - Draw a word $v_{m} \sim \text{Multi}(\pi_{zm})$

3. For each textual word $n \in \{1, \cdots, N\}$ :
   - Draw a feature index $y_{n} \sim \text{Uniform}(1, M)$
   - Draw a word $w_{n} \sim \text{Multi}(\beta_{zyn})$

Particularly, we first generate the topic proportion of visual features from either the logistic or Probit transformation of a multivariate Gaussian variable. Based on the topic proportion for each image, we generate $M$ image features conditional on the topic-feature distribution. Then for each of the $N$ textual words associated with the image, one of the image features is randomly selected to obtain the topic assignment of this image feature and next a word is generated conditional on the topic-word distribution. The corresponding probabilistic graphical model of this generative process is shown as follows.
In the context of image annotation, rather than having only the topic-word distribution, we have the topic-feature distribution acting as a bridge between the word and its corresponding topic as a topic now is actually assigned to a visual feature instead of a word.

### 4.5 Parameter Estimation

The sampling of the topic assignment for the visual features is exactly the same as equation (4) in the traditional setting of correlated topic model: this seamless extension is one of strength of the framework proposed in this thesis.

\[ P(z_{dn}^k = 1 | Z_{-n}, v_{dn}, v_{-dn}) \propto p(v_{dn} | z_{dn}^k = 1, v_{-dn}, Z_{-n})^T_k \propto \frac{C^{v_{dn}}_{k,n} + \pi_{v_{dn}}}{\sum_{j=1}^V C^{v_{dn}}_{k,j,n} + \sum_{j=1}^V \pi_j} \theta^k_d. \]  

(4.1)
where the use of $C_{-,n}$ is used to indicate that the $n$th visual word is not included in the topic or document under consideration and $\theta_d^k$ is the topic proportion based on either logistic or Probit trasnformation.

As explained in the earlier chapter, due to the non-conjucacy of posterior distribution of $\eta$, adaption of Gaussian auxilary variables is used and the matrix $Y_d \in \mathbb{R}^{N_d \times K}$ contains all the values of the auxilary variables,

$$Y_d = \begin{bmatrix}
Y_{d1}^1 & Y_{d1}^2 & \ldots & Y_{d1}^k & \ldots & Y_{d1}^K \\
Y_{d2}^1 & Y_{d2}^2 & \ldots & Y_{d2}^k & \ldots & Y_{d2}^K \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\
Y_{dN_d}^1 & Y_{dN_d}^2 & \ldots & Y_{dN_d}^k & \ldots & Y_{dN_d}^K \\
\end{bmatrix}$$

- For the component of $Y_{dn}$ whose index corresponds to the label of current topic assignment of word $n$ sample from a truncated normal distribution with variance 1 restricted to positive outcomes
  $$ (Y_{dn}^k | \eta_d^k) \sim N_+(\eta_d^k, 1) \quad z_{dn}^k = 1 $$

- For all components of $Y_{dn}$ whose indices do correspond to the label of current topic assignment of word $n$ sample from a truncated normal distribution with variance 1 restricted to negative outcomes
  $$ (Y_{dn}^l | \eta_d^l) \sim N_-(\eta_d^l, 1) \quad z_{dn}^l \neq 1 $$

Once the matrix $Y_d$ is obtained, the sampling scheme updates the parameter vector $\eta_d$ by conveniently drawing

$$ (\eta_d | Y_d, A, \mu, \Sigma) \sim MVN(\mu_{\eta_d}, \Sigma_{\eta_d}), $$
where
\[ \mu_{d} = \Sigma_{d}(\Sigma^{-1} \mu + X_{d}^{T}A^{-1}\text{vec}(Y_{d})) \quad \text{and} \quad \Sigma_{d} = (\Sigma^{-1} + X_{d}^{T}A^{-1}X_{d})^{-1}. \]

with \( X_{d} = 1_{n \times d} \otimes I_{K} \) and \( \text{vec}(Y_{d}) \) representing the row-wise vectorization of the matrix \( Y_{d} \). Adopting the fully Bayesian treatment of our Probit normal correlated topic model, we once again add an extra layer to the hierarchy in order to capture the variation in the mean vector and the variance-covariance matrix of the parameter vector \( \eta_{d} \). Taking advantage of conjugacy, we specify a normal-Inverse-Wishart prior for \( (\mu, \Sigma) \), namely,
\[ p(\mu, \Sigma) = NIW(\mu_{0}, \kappa_{0}, \Psi_{0}, \nu_{0}), \]
meaning that \( \Sigma|\nu_{0}, \Psi_{0} \sim IW(\Psi_{0}, \nu_{0}) \) and \( (\mu|\mu_{0}, \Sigma, \kappa_{0}) \sim MVN(\mu_{0}, \Sigma / \kappa_{0}) \). The corresponding posterior is normal-inverse-Wishart, so that we can write
\[ p(\mu, \Sigma|W, Z, \eta) = NIW(\mu', \kappa', \Psi', \nu'), \]
where \( \kappa' = \kappa_{0} + D, \nu' = \nu_{0} + D, \mu' = \frac{D}{D + \kappa_{0}} \bar{\eta} + \frac{\kappa_{0}}{D + \kappa_{0}} \mu_{0}, \text{and} \)
\[ \Psi' = \Psi_{0} + Q + \frac{\kappa_{0}}{\kappa_{0} + D} (\bar{\eta} - \mu_{0})(\bar{\eta} - \mu_{0})^{T}, \]
where
\[ Q = \sum_{d=1}^{D}(\eta_{d} - \bar{\eta})(\eta_{d} - \bar{\eta})^{T}. \]

A closer look at the probabilistic graphical model reveals that all the sampling schemes up to this point are exactly identical to the original Probit normal model except that words in the previous case is now visual words. However, the sampling scheme for the index
variable \( y_n \) needs to be developed to complete the full conditional distribution for the Gibbs Sampler and is shown as below:

\[
p(y_{m}^{v_i} = k | w_n^y = w, y_{-m}^{v_i}, w_{-n}^{v_i}, z_{v_i}) \propto \frac{C_{k,v_i}}{\sum_k C_{k,v_i}} \frac{C_{k,n}^{w_n} + \beta}{\sum_j C_{k,-n}^{w_j} + V \beta}
\] (4.2)

Where \( C_{k,v} \) represents the total number of times topic \( k \) is assigned to visual feature \( v_i \) and \( C_{k,-n}^{w_n} \) is the total number of words assigned to topic \( k \). This distribution is intuitive and consists of two terms. The first term measures the likelihood of assigning topic \( k \) to word \( w_n \) through the topic assignment of its corresponding \( v_i \) since the word is assigned to a visual feature and then linked with the topic assignment of that visual feature. The second term represents the probability of generating word \( w_n \) from topic \( k \).

For the test images, each topic proportion \( \theta_d^k \) is computed by running Gibbs Sampling again for unseen images with the topic word distribution obtained from the training phase. Finally, the total probability that an image \( I \) is annotated with word \( w_n \), can be calculated as follows:

\[
p(w_n | I) = \sum_k p(k | I) p(w_n | k) = \sum_k \theta_d^k \beta_k^{w_n}
\] (4.3)

Where \( \theta_d^k \) is the posterior topic proportion for the visual assignments extracted from the images and \( \beta_k^{w_n} \) is the probability of seeing word \( w_n \) when topic \( k \) is active. Then we rank the probability \( p(w_n | I) \) for all \( n \) words and choose the words with the highest total probability to be the caption of its corresponding image.
4.6 Computational Results

For each image in the BBC dataset, 100 to 125 SIFT features are extracted due to the memory issue encountered in the K means clustering algorithms, generally if we have sufficient computer memory, we would not restrict the maximum SIFT features extracted from an image. These SIFT features were quantized and grouped into a discrete set of visual words using K-means with 600 clusters. Once again due to the memory issue of the computer, 600 clusters is the maximum number we could reach. We varied \( K \) from 500 to 1000 and set prior parameter for \( \beta = 0.001 \) and \( \pi = 0.01 \) for all the settings.

The following examples are the results for the image annotation using Probit Normal Correlated Topic Model:

![Image: Annotation Results](image)

"academic", "school", "cost", "building", "education"

_Figure 16 Annotation Results_
"space", "moon", "orbit", "mission", "scientist"

"canadian", "afghanistan", "kandahar", "mission"

"instrument", "musician", "hold" "cabin", "luggage"
“zarqawi”, "caldwel", "death", “iraqi”, "suggest"

"reid"

"shia", "darfur", "iraqi", "mogadishu", "state", "oil"

"cloud", "space", "product", "airbus"

"flight", "aircraft", "cia", "plane", "traffic", "airport"
As can be seen from the above results, human inspection reveals that the proposed scheme does a very decent job at annotating unseen images.

4.7 Conclusion and Future Work

The comparison of different algorithms in terms of the accuracy of image retrieval and annotation is beyond the scope of the thesis and will be presented as a future work subsequently. The reason for this lies in the fact such assessment requires computer resources that we do not have in our disposal with the time constraint of this thesis. As stated in the earlier sections, the BBC dataset used for the image annotation task is very noisy and often the descriptive articles does not directly reflect the actual objects presented in the image. Therefore, future study and comparison across algorithms will be focusing on some cleaner datasets especially the Corel 5k dataset and ImageNet dataset in which each image is only paired with the most descriptive words. In addition, the extraction of visual features plays a vital role in the task and image annotation and some researches have indicated that K means is not a robust way of clustering visual images. Instead, methods such as [46, 47] should be used to group image features to its associated category.

Scale Invariant Feature Transform (SIFT) is the major feature extraction techniques employed in this thesis for image annotation, however, there are many other techniques available to represent an image as Bag-of-Features. Other influential feature descriptors include techniques such as Speeded Up Robust Features (SURF) [48], Oriented FAST and Rotated BRIE (ORB) [49] and Histogram of Oriented Gradients (HOG) [50]. Therefore, in the context of topic modeling, we could compare the annotation result
obtained from each of the feature extraction technique. Aside from solely relying on the feature descriptors as the input for an image, we could incorporate MSER region detector into our model by having an enriched representation of an image. [36] have shown that the addition of extra layer using Maximally stable extremal regions (MSER) region detector greatly improve the annotation task in the context of image annotation using topic modeling.
Chapter 5

5. High performance Computing

In this chapter, we introduce the high performance computing methods employed in our implementation of our proposed models to realize our goal of developing an efficient and scalable algorithm. In the context of topic modeling, it is inevitable to utilize various high performance computing techniques to be able to generate model output in a timely fashion. Techniques used in this thesis includes the implementation of the framework and architecture of Map Reduce, vectorization of sampling, memory aware representation of variables and various numerical efficiency strategies. The main computing program used in this thesis is the R language which is one of the most popular and powerful statistical software. Thanks to the recent development of “Rcpp” package, a great number of R functions now are wrapped with R while running in C in the background, which greatly improves the computational efficiency while maintaining the user-friendly interface of R. Therefore we illustrate the high performance computing in the R computing environment with a 32-core Linux work station in this chapter. We will refer to each of the core from the 32-core workstation as a worker or processor interchangeably. And for each sampling task we parallelize to the workers, we called them jobs.

5.1 Distributed Topic Modeling via Map Reduce

Map Reduce processes large data set with a parallel distributed architecture to improve the computational efficiency of an algorithm through a cluster of computers or a single computer with multiple cores. With today’s large scale of the data, applications such as
topic models needs to infer model parameters in a scalable fashion to keep up with constant update of new information. Therefore, a computationally efficient system is much in need to implement these sophisticated models. Since the introduction of Latent Dirichlet Allocation from a decade ago, a series of excellent distributed algorithms have been developed to improve the efficiency of the algorithm. [30] developed a so-called “embarrassingly” parallel algorithm through a large number of clusters. [29] developed a time and memory efficient Gibbs Sampling called SparseLDA. [51] developed a fast Metropolis Hasting algorithm called AliasLDA to allow topic assignment to be sampled with $O(1)$ time complexity by re-using a pre-computed alias table over many tokens. Recently [18] combined the advantages from all the previous methods and created a structure-aware model parallel scheme with an improved Metropolis Hastings sampling algorithm that is invariant to the model size and converges an order of magnitude faster than the state of art Gibbs Sampling algorithms. In this thesis, Gibbs Sampling is the major techniques employed and hence we adapted the methods from [29] and [30] to improve the computational efficiency by implementing these distributed algorithms to a 32-core Linux Workstation. The latter two methods involving Metropolis Hasting is beyond the scope of this thesis and will be incorporated in our future work.

Firstly, I introduce a motivating example for a simple distributed algorithm for sampling a Gaussian distributed variable many times, which offers insight for the reasons why distributed algorithms are attractive when handling large dataset and complex algorithms. Imagine one has to obtain 100,000 samples, in R we could use the command “rnorm(100000)” to obtain the samples. However for fair comparison and the purpose of illustration, we sample “rnom(1)” for 100,000 times in the regular setting, while in the
parallel setting, we distributed the sampling of 100,000 “rnorm(1)” to our 32-cores Linux workstation. I repeated each process for 30 times and the result is shown in Figure 17.

![Figure 17 Efficiency Comparisons between Parallel and Non-Parallel](image)

Based on this motivating example, it is evident that by simply parallelizing the jobs into each of the worker or core, the efficiency gained in terms of time is clear. This is a straightforward version of Map Reduce known as the “embarrassingly” parallel algorithm in which all we do to improve the efficiency is dividing the jobs at hand into pieces and then feeding them to each worker to process. When all the workers finish processing the jobs, we aggregate the results according to the ordering of the jobs. This embarrassingly parallel algorithm is shown in Figure 18.
However in most interesting applications, using the embarrassingly parallel implementation is far from enough and in some cases may be very suboptimal. Since the jobs are dispatched to workers, we need to create local copy of all the variables and data needed to process the job. Therefore, information exchange between workers have to be kept at minimal and the storage of variables needs to be efficient because the overhead of the information exchange is very expensive and in industry size clusters, high bandwidth wires are preferred to allow fast information transfer between cluster nodes.

From section 2.2.3, we know the posterior inference for Latent Dirichlet Allocation came down to computing the conditional distribution of topic assignment for each word:

\[
p(z_i | z_{-i}, \alpha, \beta, w) \propto (n_{a,k} + \alpha_k) \frac{n_{k,w} + \beta_w}{n_{k,\cdot} + \beta_{\cdot}} \]

Unfortunately, computing the topic assignment for each word in the corpus sequentially is computationally impossible for large dataset and may even take months to complete for large dataset. Hence, we can implement the embarrassingly parallelization [30] by
sending a subset of the documents to each worker in a block so that documents could be processed independently instead of sequentially. The pipeline of this simple parallel LDA is shown in Figure 19.

![Figure 19 Parallel Implementation of LDA [31]](image)

Similar to LDA, we adapted similar implementation into our proposed Probit Normal Correlated Topic Model to compute the topic assignment independently across the documents. However, our model allows an enriched representation of topical relationship by the introduction of Gaussian random variable $\eta$. As explained in section 3.3, the sampling of $\eta$ relies on another Gaussian distributed auxiliary variable $Y$. Hence we present a block parallel sampling by sampling these two variables together for each distributed documents as $\eta$ depends only on the auxiliary variable $Y$ while $Y$ depends on the topic assignment $z$ from the previous iteration. This block parallel sampling minimizes the information passing between workers which decreases the memory consumption. As a result, our parallel implementation includes two block samplers in which $Y$ and $\eta$ are sampled together as a block while topic assignment $z$ follows the same procedure as LDA.

Due to various reasons including disk accesses, different job load (the length of each document varies) and overhead time, the algorithm has to wait for the slowest processor
to finish the job and then aggregate all the updated topic assignment. Therefore, function such as “sfClusterApplyLB” from the “snowfall” package in R provides an excellent solution by allowing a dynamic scheduling of jobs. When a worker completes the entire scheduled task, unfinished jobs from other workers are rescheduled and therefore decreased the idle time by load balancing.

Another common problem for multiprocessor is the memory consumption as multicore systems automatically lead to an $O(\text{processor})$ increase in the allocated memory as local copy of variables and dataset needs to be created. As a result, memory outage will cause the entire algorithm to freeze.

Therefore, the adaptation of the Sparse-LDA implementation of the Gibbs Sampler not only relieves this problem but also decreases the sampling complexity. Specifically we illustrate this method in the sampling scheme of our proposed model, which follows a similar formulation from [4]

Firstly equation (4.1) can be rewritten as:

$$P(z_{dn}^k = 1|Z_{-n}, w_{dn}, W_{-dn}) = \frac{A_k}{S_A + S_B} + \frac{B_k}{S_A + S_B}$$

Where

$$A_k = \frac{c_{k,-n}^{w_{dn}}}{\sum_{j=1}^{V} c_{k,-n}^{j} + \nu \beta_j} \theta_d^k, \quad B_k = \frac{\beta_{w_{dn}}}{\sum_{j=1}^{V} c_{k,-n}^{j} + \nu \beta_j} \theta_d^k, \quad S_A = \sum_k A_k, \quad S_B = \sum_k B_k$$

By the formulation in equation , sampling of topic assignment $z_{dn}$ can be sampled from Multinomial($\frac{A}{Z_A}$) or Multinomial($\frac{B}{Z_B}$)

$$P(z_{dn}^k = 1|Z_{-n}, w_{dn}, W_{-dn}) = \frac{A_k}{S_A + S_B} + \frac{B_k}{S_A + S_B} = \frac{(1 - p)A_k}{S_A} + \frac{pB_k}{S_B}$$
Where \( p = \frac{S_B}{S_A + S_B} \) and a closer look at this formulation reveals that it is a marginalization with respect an auxiliary binary variable. Therefore, \( z_{dn} \) can be sampled by simply examining the outcome of flipping a coin with probability \( p = \frac{S_B}{S_A + S_B} \) being head. If the outcome is head, \( z_{dn} \) is sampled from Multinomial\( \left( \frac{B}{Z_B} \right) \), otherwise from Multinomial\( \left( \frac{A}{Z_A} \right) \).

The efficiency of this particular method stems from the sparsity of matrix \( A \) and in fact \( A \) is very sparse since a word is normally assigned to a very small subset of topics in the entire topic space. Due to this sparsity, the time complexity of the sampling of topic assignment \( Z_{dn} \) could be reduced from \( O(K) \) to \( O(s(K)) \), where \( s(K) \) represents the expected number of non-zero entries in the count matrix \( C_k \). In fact, the count matrix \( C_k \) is very sparse and become even more sparse as the Markov Chain approaches to a stationary distribution. Therefore in practice, \( s(K) \ll K \) when the topic space is relatively large. In applications such as implementing our proposed model on image annotation, the topic space will be very large to accommodate for visual features. Thus the gain in efficiency is very good with the SparseLDA representation. On the other hand, we sample from Multinomial\( \left( \frac{B}{Z_B} \right) \), we vectorized over all \( K \) to obtain the topic assignment. Once again thanks to the sparsity, \( O(K) \) is reasonable.

### 5.2 Vectorization and Efficient Memory Usage

In this section, we present two general techniques used in the implementation of our proposed model to further increase the computational performance of the algorithms. These two general techniques also could be used in all modeling situations since they greatly
decrease the computational cost and memory consumption by simply changing the way of writing computer codes.

5.2.1 Efficient Memory Usage

As described in the previous section, efficient memory usage is crucial during the implementation of parallel algorithms since multicore systems automatically lead to an $O(p)$ increase in the allocated memory as local copy of variables and dataset needs to be created. Taking the topic assignment $Z_{wn}$ for word $w_n$ as an example, $Z_{wn}$ is a $K$ (total number of topics) dimensional binary vector in which the $i^{th}$ element is 1, meaning $w_n$ is assigned to topic $i$. Therefore, the topic assignment matrix $Z_d$ for document $d$ is a $K$ (total number of topics) by $N_d$ (total number of words in document $d$) in which only $K$ number of the entry is non-zero. For instance, for the BBC dataset we used in chapter 3 and 4, we have 135 words on average for each document and a fixed $K = 1000$ for modeling the topical space. If we simply store $Z_d$ as a $K$ by $N_d$ matrix without awareness of the memory consumption, we will have a total number of 3121 matrices with this size for the entire corpus, which in total adds up to 3214 Megabytes or 3 Gigabytes of memory space. In the parallel setting with a regular 4-cores computer with 8 Gigabytes memory, passing these matrix to all the cores will take up to 12 Gigabytes of memory which well exceeds the total amount of the memory space. However, it is inevitable to have this $K$ by $N_d$ matrix for each document in the calculation of $C_{k^i}$ as vectorized calculation perform much faster than repeated structure using for loops, which we will demonstrate in the next section. Therefore, we could simply store the index of the topic
assignment instead of a $K$ dimensional binary vector. For example, suppose we have a document with 6 words and the topic assignment of this document is shown in Table 5.

<table>
<thead>
<tr>
<th>Topic 1</th>
<th>cat</th>
<th>cute</th>
<th>dog</th>
<th>eat</th>
<th>fruit</th>
<th>vegetable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topic 1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Topic 2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Topic 3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Instead of storing the topic assignment as a matrix, we could simply create a vector (2,2,2,3,1,1) which points to the topic index of each word in the document. Storing the topic assignment in this way will decrease the total size from 3 Gigabytes to merely 3.3 Megabytes for the entire corpus. In the vectorized sampling for the new topic assignment, we could reconstruct the topic assignment matrix again by using the “sparseMatrix” function and therefore having at most 32 matrices in the parallel implementation on a 32-core computer. In the original scheme where the topic assignment matrix is stored as its own, we created 32 local copies of 3121 matrices which leads to 99872 matrices stored on the entire computer.

### 5.2.2 Vectorization

Vectorization is one of the most important and fundamental concept in high performance computing since it is easy to be implemented and the gain from vectorization is remarkable. In most languages such as R and Python, for loops are generally computationally inefficient and needs to be avoided by all means if possible. Even for object oriented language C and C++ where the compiler compiles the for loops and implement it in a vectorization fashion, vectorization is still be able to improve the
computational cost. Throughout the implementation of our proposed model in this thesis, except for the progress of each iteration, all the for loops are replaced with a fast vectorized calculation of the sampling scheme of interest. We herein give a short demonstration of vectorization the calculation of topic proportion $\theta_d$ for document $d$. Since we parallelized the documents to workers for independent processing. Thus within each worker, we would not be able to parallelize the calculation for each component of the topic proportion $\theta_d^k$ again.

$$\theta_d^k = \frac{ \left( 1 - \Phi(-\eta_d^k) \right) \prod_{j \neq k} \Phi(-\eta_d^j) }{ \sum_{j=1}^{K} \left( 1 - \Phi(-\eta_d^j) \right) \prod_{j \neq k} \Phi(-\eta_d^j) }$$  \hspace{1cm} (5.2)

Based on the equation (5.2), each component of the topic proportion $\theta_d^k$ is computed individually by multiple the product of all the $\Phi(-\eta_d^j)$ for all non-$k$ position with $\left( 1 - \Phi(-\eta_d^k) \right)$ for position $k$ and then normalized to sum to 1. However, computing each component sequentially is expensive under each worker and thus vectorization calculation could be performed via the following steps in which we disregard the position index first to calculate the product of the probability density and later dividing its corresponding component with index $k$

1. `fast<-pnorm(-prior[ta[D,]])`
2. `thetakd<-(1-fast)*prod(fast)/fast`

With 500 topics, this simple vectorization decreases the time for sampling the topic proportion $\theta_d$ for the entire corpus from 18.86 seconds to 0.065 seconds for each iteration. With the same setting of 500 topics, sampling the topic assignment using vectorization method decreases the time from 5 minutes to mere 10 seconds for the BBC data set.
5.3 Numerical Efficiency Strategies

In this section, I explain several numerical efficiency strategies used in this thesis to improve some expensive operations such as matrix inversion of large matrix and computation of kronecker product.

5.3.1 Cholesky Decomposition for Large Matrix Inversion

In the sampling step for parameter \( \eta \sim (\eta_d | Y_d, A, \mu, \Sigma) \sim MVN(\mu_{\eta_d}, \Sigma_{\eta_d}) \), where

\[
\mu_{\eta_d} = \Sigma_{\eta_d}^{-1} \mu + X_d^T A^{-1} vec(Y_d) \quad \text{and} \quad \Sigma_{\eta_d} = (\Sigma^{-1} + X_d^T A^{-1} X_d)^{-1}.
\]

There are multiple high dimensional matrix inversion involved in this sampling and hence Cholesky decomposition could be used to accelerate this process and the actual improvement is actually twice as fast as the traditional matrix inversion. We are currently working on alternative ways to completely avoid calculating the matrix inversion in our algorithm as in the context of topic modeling, it is inevitable to have an extremely large topic space.

5.3.2 Simplifying the Kronecker Product

Again in the sampling step for parameter \( \eta \sim (\eta_d | Y_d, A, \mu, \Sigma) \sim MVN(\mu_{\eta_d}, \Sigma_{\eta_d}) \), where

\[
\mu_{\eta_d} = \Sigma_{\eta_d}^{-1} \mu + X_d^T A^{-1} vec(Y_d) \quad \text{and} \quad \Sigma_{\eta_d} = (\Sigma^{-1} + X_d^T A^{-1} X_d)^{-1}.
\]

The kronecker product associated has a size of 1 Gigabytes for each document in the corpus, therefore for the BBC data in which we have over 3000 documents, Kronecker product will freeze the algorithm in large topical space. Nevertheless, \( X_d^T A^{-1} vec(Y_d) \) could be simplified to taking the column sum of matrix \( Y_d \) when \( A \) is a diagonal matrix. Since the introduction
of \( \eta \) already brings in the correlation structure among topics, it would be unnecessary to add an extra layer of correlation in \( A \). Therefore, the kronecker product could be removed by simply calculating the column sum of matrix \( Y_d \).
Chapter 6

6. Conclusion and Future Work

In this thesis, we explored topic models from various perspectives ranging from introducing different types of topic models to the scalability of the modeling. More importantly, we developed an alternative Correlated Topic Model called Probit Normal Correlated Topic Model by adapting an recent advancement of the Multinomial Probit Model [7] and created a Gibbs sampling schemes in the context of image annotation for both the traditional correlated topic model and our proposed model. With successful results obtained for both documents and images, we demonstrated the strength and potential of our model and contributed to the topic modeling research community. In addition, we incorporated various high performance techniques including memory-aware Map Reduce, SparseLDA implementation, vectorization and various state-of-the-art numerical efficiency strategies to allow scalable sampling and computation.

In section 3.5 of chapter 3 and section 4.7 of chapter 4, we outlined a series of possible future works from both the scalability and modeling perspective.

Over the course of completing this master thesis, I have gained tremendous insight into various aspects of modern statistics in the age of big data and I feel very fortunate that my advisor Dr. Fokoue recommended me to work in this field. The completion of this work will be a stepping stone for me to climb for a higher mountain during my PhD study at UC Santa Cruz.
Bibliography


http://dx.doi.org/10.1080/01621459.1993.10476321


Appendix

SIFT Implementation via Python

###Import Package###

```python
import cv2
import numpy as np
import os
import glob
import scipy.cluster
```

```python
os.chdir('default')
images = []

###Read in all the images in gray scale###

```python
for infile in glob.glob('./*.jpg'):
    pic = cv2.imread(infile,0)
    images.append(pic)
```

```python
###Initialization of variables###

```python
my_set = images
descriptors = np.array([])
feaL=np.array([])
```

```python
###Apply SIFT algorithm to all images###

```python
for pic in my_set:
    kp, des = cv2.SIFT(125).detectAndCompute(pic, None)
    feaL=np.append(feaL,des.shape[0])
```
descriptors = np.append(descriptors, des)

### Organize vector to a n by 128 matrix ###
desc = np.reshape(descriptors, (len(descriptors)/128, 128))
desc = np.float32(desc)

from scipy.cluster.vq import whiten
desc=whiten(desc)

### K Means Clustering with 600 groups ###
kclustere=600

codebook=scipy.cluster.vq.kmeans(desc, k_or_guess=kclustere, iter=20, thresh=1e-05)

### Obtain centroids from the K means ###
label=scipy.cluster.vq.vq(desc,codebook[0])

### Represent visual featrues as bag-of-feature ###
i=0
fvr=0
FeatureCode=()for index in range(0,feaL.shape[0]):
    i=feaL[index]
    fvr=np.bincount(label[0][sum(feaL[0:index]):sum(feaL[0:(index+1)])])
    print(label[0][sum(feaL[0:index]):sum(feaL[0:(index+1)])].shape)
    print(len(fvr))
    FeatureCode=np.append(FeatureCode,fvr)

Feature=FeatureCode.reshape(len(feaL),1,kclustere)