Fully Bayesian Inference for Finite and Infinite Discrete Exponential Mixture Models

Xuanbo Su

A Thesis

in

The Concordia Institute

for

Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of

Master of Applied Science (Quality Systems Engineering) at

Concordia University

Montréal, Québec, Canada

December 2021

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: **Xuanbo Su**

Entitled: **Fully Bayesian Inference for Finite and Infinite Discrete Exponential Mixture Models**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science (Quality Systems Engineering)

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the Final Examining Committee:

_____ Chair
Dr. Zachary Patterson

_____ Examiner
Dr. Jamal Bentahar

_____ Supervisor
Dr. Nizar Bouguila

_____ Co-supervisor
Dr. Nuha Zamzami

Approved by _____
Abdessamad Ben Hamza, Chair
Concordia Institute for Information Systems Engineering

_____ 2021

_____
Mourad Debbabi, Dean
Faculty of Engineering and Computer Science

Abstract

Fully Bayesian Inference for Finite and Infinite Discrete Exponential Mixture Models

Xuanbo Su

Count data often appears in natural language processing and computer vision applications. For example, in images and textual documents clustering, each image or text can be described by a histogram of visual words or text words. In real applications, these frequency vectors often show high-dimensional and sparsity nature. In this case, hierarchical Bayesian modeling frameworks show the ability to model the dependence of the word repetitive occurrences 'burstiness'. Moreover, approximating these models to exponential families is helpful to improve computing efficiency, especially when facing high-dimensional count data and large data sets. However, classical deterministic approaches such as expectation-maximization (EM) do not achieve good results in real-life complex applications. This thesis explores the use of a fully Bayesian inference for finite discrete exponential mixture models of Multinomial Generalized Dirichlet (EMGD), Multinomial Beta-Liouville (EMBL), Multinomial Scaled Dirichlet (EMSD), and Multinomial Shifted Scaled Dirichlet (EMSSD). Finite mixtures have already shown superior performance in real data sets clustering with EM approach. The proposed approaches in this thesis are based on Monte Carlo simulation technique of Gibbs sampling mixed with Metropolis-Hastings step, and we utilize exponential family conjugate prior information to construct the required posteriors relying on Bayesian theory. Furthermore, we also present the infinite models based on Dirichlet processes, which results in clustering algorithms that do not require the specification of the number of mixture components to be given in advance. The performance of our Bayesian approaches was tested in some challenging real-world applications concerning text sentiment analysis, fake news detection, and human face gender recognition.

Acknowledgments

First of all, I am very thankful to my supervisor, Nizar Bouguila, for giving me a lot of help and guidance on my thesis. He provides me the independence to do research and teach me how to use Bayesian theory in machine learning.

I am also grateful to professor Nuha Zamzami, who not only provided me with ideas to solve the problems I encountered in the project but also offered a lot of suggestions on experimental choices. Frankly, I could not complete this thesis without the support of Nizar Boouguilia and Nuha Zamzami.

In addition, I would like to thank Dr. Najar Fatama, who gave me a lot of help and detailed advice in the infinite model section. I would also like to thank my friends and classmates, especially Bing-Wei Ge and Tiqanqi Yang, who gave me a lot of project advice and life support. Moreover, I want to express my sincere thanks to my parents for their spiritual and financial support throughout my graduate studies term.

Finally, I'd want to express my gratitude to Concordia University for providing me with a comfortable and beautiful campus environment, particularly the library, which is available 24 hours a day.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

As technology advances, more and more complicated data are generated. Evaluating such valuable data and extracting latent pattern is a topic of interest in a variety of scientific and technological fields. One of the main attention grabbing approaches is clustering, and finite mixture models have been frequently used to cluster data into homogeneous groups because of their flexibility and ease of use. Clustering count vectors is a challenging task on large data sets considering its high dimensionality and sparsity nature [38]. Bag of words representation for text systematically exhibits the burstiness phenomenon, if a word appears once in a document, it is much more likely to appear again [20, 39]. This phenomenon is not limited to text and can also be observed in images with visual words [36]. It also has the sparsity nature that few words show up with high occurrence and some are less as often as possible or do not appear at all [43]. Thus, such data are generally represented as sparse high-dimensional vectors, with few thousands of dimensions with a sparsity of 95% to 99% [25]. Hierarchical Bayesian modeling frameworks, such as Multinomial Generalized Dirichlet mixture model (GDM), Multinomial Beta-Liouville mixture model (MBL), Multinomial Scaled Dirichlet (MSD), and Multinomial Shifted-Scaled Dirichlet mixture model (MSSD) [2, 9, 30, 51] , have shown excellent performance for high-dimensional count data clustering with Expectation-maximization (EM) approach. However, their estimation procedures are very inefficient when the collection size is large [60]. The exponential family of distributions

has a finite-sized sufficient statistics [16], meaning that we can compress the data into a fixed-sized summary without loss of information [24]. Efficient exponential-family approximations to the MGD (EMGD), MBL (EMBL), MSSD (EMSSD), and MSD (EMSD), have been previously proposed by Zamzami and Bouguila [60–62]. These distributions have been shown to address the burstiness phenomenon successfully and to be considerably computationally faster than their original distribution forms especially when dealing with sparse and high-dimensional data. The main problem in the case of finite mixture models is the estimation of the model parameters [15]. EM algorithm is a simple and effective approach for model parameters estimation [41]. However, the EM algorithm for finite mixtures has several drawbacks. For example, the occurrence of local maximum and singularities in likelihood function will often cause problems for deterministic gradients method [48]. Moreover, in high dimensional estimation, it will be hard to obtain reliable estimates which possess generalization capabilities to predict the densities at new data points [18, 26]. Some Bayesian approaches are based on simulation methods, such as Gibbs sampling, which explore high-density regions [3, 29]. The stochastic aspect of these simulation methods ensures the escape from local maximum [13]. Tsionas [53] proposed an estimation approach for multivariate t distribution using Gibbs sampling with data augmentation. Amirkhani, Manouchehri, and Bouguila [4] presented a fully Bayesian approach within Monte Carlo simulation for Multivariate Beta mixture parameters estimation. Bouguila, Ziou, and Hammoud [13] successfully adopted a Bayesian algorithm based on Metropolis-within-Gibbs sampling for a finite Generalized Dirichlet mixture. Najar, Zamzami, and Bouguila [45] used Monte Carlo simulation method for exponential family approximation to the Dirichlet Compound Multinomial mixture model (EDCM) parameters estimation and showed excellent results in some real applications. Another challenging aspect when using a finite mixture model is usually to estimate the number of clusters which best describes the data without overfitting or underfitting it. For this purpose, many approaches have been suggested. These approaches can be divided into two different ways for mixture models. The first way is the implementation of model selection criteria. The second way is resampling from the full posterior distribution with the number of clusters considered unknown. However, the majority of these approaches cannot be easily used for high-dimensional data [12]. The infinite mixture models based on Dirichlet process have recently attracted wide attention, thanks to the development of MCMC techniques. Dirichlet process

(DP) will resolve the difficulties related to model selection. Rasmussen [47] successfully applied Dirichlet process on Gaussian mixture model with Gibbs sampling to obtain accurate number of classes. Bouguila and Ziou [12] also presented a clustering algorithm for Dirichlet process mixture of Generalized Dirichlet distributions with MCMC techniques. Najar, Zamzami, and Bouguila [46] proposed an infinite mixture of exponential family approximation to the Multinomial Dirichlet Compound mixture model and shown superior experimental results in recognition of human interactions in feature films. Thus, we extend these finite mixture models to infinite mixture models based on Dirichlet process to tackle model selection in the case of sparse high-dimensional vectors.

1.2 Literature Review and Background

1.2.1 Monte Carlo Approximation

If we have enough samples from a distribution p, then according to the law of large numbers, the average value will converge to the expected value. The estimator is given by [8]:

$$I = \frac{1}{N}\sum_{i=1}^{N} f(x_i) \approx E_p[f(x)] \tag{1}$$

where we take N samples, $x_1, \cdots x_n$, from the p distribution. However, it has some obvious disadvantages that it is computationally expensive and there are some distributions that we cannot sample the data directly from.

1.2.2 Importance Sampling

This approach adds a proposal distribution based on Monte Carlo approximation to solve the problem of not being able to sample directly from the original distribution. We choose a proposal distribution that matches the shape of the target distribution and is easier to sample from, then, we can compute the expectation [8]:

$$E_p[\frac{f(x)}{q(x)}] = \int \frac{f(x)}{q(x)} q(x) d_x \approx \frac{1}{N}\sum_{i=1}^{N} \frac{f(x_i)}{q(x_i)} \tag{2}$$

1.2.3 Gibbs Sampling

The Gibbs sampler is a technique for generating random variables from a multivariate (marginal) distribution indirectly, without having to calculate the density [8]. Define a multivariate distribution as $p(x_1, \cdots, x_n)$. Then, the Gibbs sampling algorithm can be summarized as [33]:

Algorithm 1 Gibbs sampling

Input: $x^0 = x_1, \cdots, x_N, K$
for $t = 1 \rightarrow K$ **do**
$x_1^t \sim p(x_1 | x_2^{t-1}, \cdots, x_N^{t-1})$
$x_2^t \sim p(x_2 | x_1^t, x_3^{t-1} \cdots, x_N^{t-1})$
$\cdots$
$x_{N-1}^t \sim p(x_{N-1} | x_1^t, \cdots, x_{N-2}^t, x_N^{t-1})$
$x_N^t \sim p(x_N | x_1^t, \cdots, x_{N-1}^t)$
end for

where x^0 is initial vector, and K is the sampling number.

1.2.4 Metropolis-Hastings Sampling

Metropolis, Rosenbluth, and Teller proposed the basic Metropolis algorithm [44], which was then popularized by Hastings to Metropolis-Hastings (M-H) algorithm [35]. This algorithm is very general, it can sample data from a complicated distribution, it also needs to choose a proposal distribution q.

Algorithm 2 M-H sampling

Input: x^0, K, q
for $t = 1 \rightarrow K$ **do**
    $x^{'} \sim q(x^{'} | x^t)$
    $A(x^{'} | x) = (1, \frac{p(x^{'}) q(x | x^{'})}{p(x) q(x^{'} | x)})$
    $u \sim [0, 1]$
    if $u <= A(x^{'} | x)$ **then** $x_{t+1} = x^{'}$ **else** $x_{t+1} = x_t$
    end if
end for

The obvious drawback of this algorithm is that it is very dependent on the proposal distribution, which will affect the convergence speed and even the final results.

1.2.5 Finite Mixture Model

A finite mixture of distributions with M component is defined as [32]:

$$P(X|\Theta) = \sum_{j=1}^{M} p(X|\xi_j)P_j \tag{3}$$

where the P_j are the mixing weights and $p(X|\xi_j)$ is the components distribution, $\Theta = (\xi, P)$ is the entire set of parameters to be estimated, where $\xi = (\xi_1 \cdots \xi_M)$, ξ_j is represents the parameters of distribution j, and $P = (P_1, \cdots, P_M)$ is the vector of mixing weights [5] and it must satisfy: $0 \leq P_j \leq 1, j = 1 \cdots M, \sum_{j=1}^{M} P_j = 1$.

1.2.6 Exponential Family

The exponential family of distributions is widely used in machine learning research due to its sufficient property, as the sufficient statistics can give all of needed parameter information by the whole sample data set [37]. For a random variable X and a distribution with M parameters in exponential-family we have [32]:

$$p(X|\xi) \propto H(X)exp(\sum_{l=1}^{M} G_l(\xi)T_l(X) + \Phi(\xi)) \tag{4}$$

where $G_l(\xi)$ is called the natural parameter, T_l(X) is the sufficient statistic, $H(X)$ is the underlying measure, and $\Phi(\xi)$ is called log normalizer used to ensure that the distribution integrates to one [24].

1.2.7 The Exponential Family Approximation to Multinomial Generalized Dirichlet Distribution

The Dirichlet assumption imposes a negative-correlation requirement where all variables in a random vector are restricted to be negatively correlated. Thus, it is not possible to include a specific variance information for each entry of the random vector [17]. Moreover, additional strenuous constraints are set on the variances and the covariances in case of using the mean probabilities to solve the parameters of a Dirichlet distribution [55]. The equal-confidence condition is another constraint of the Dirichlet distribution [56]. A random variable with a small normalized variance is generally

less uncertain than one with a sizable normalized variance. However, in a Dirichlet random vector, the normalized variance is the same for all variables. The Generalized Dirichlet distribution (GD) can release the constraints of variance information in Dirichlet distribution [27]. Moreover, The independence property of GD distribution, characterized by the ability to sample every single entry of the random vector from an independent Beta distribution gives more modeling flexibility, compared with the Dirichlet distribution [17].

Define $\rho = (\rho_1, \cdots, \rho_D)$, the GD distribution with parameters $\alpha = (\alpha_1, \cdots, \alpha_D)$ and $\beta = (\beta_1, \cdots, \beta_D)$, is defined as [54]:

$$GD(\rho|\alpha, \beta) = \prod_{d=1}^{D} \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} \rho_d^{\alpha_d - 1} (1 - \sum_{l=1}^{d} \rho_d)^{\eta_l} \tag{5}$$

where $0 < \rho_d < 1, \eta_l = \beta_d - \alpha_{d+1} - \beta_{d+1}$, for $d = 1, \cdots, D-1$ and $\eta_D = \beta_d - 1$

The mean and variance of the GD distribution satisfy the following [54]:

$$\begin{aligned} E(\rho_d) &= \frac{\alpha_d}{\alpha_d + \beta_d} \prod_{l=1}^{d-1} \frac{\beta_l}{\alpha_l + \beta_l} \\ Var(\rho_d) &= E(\rho_d)(\frac{\alpha_d + 1}{\alpha_d + \beta_d + 1} \prod_{l=1}^{d-1} \frac{\beta_l + 1}{\alpha_l + \beta_l + 1} - E(x_d)) \end{aligned} \tag{6}$$

and the convariance between ρ_{d1} and ρ_{d2} is:

$$Cov(\rho_{d1}, \rho_{d2}) = E(\rho_{d2})(\frac{\alpha_{d1} + 1}{\alpha_{d1} + \beta_{d1} + 1} \prod_{l=1}^{d1-1} \frac{\beta_l + 1}{\alpha_l + \beta_l + 1} - E(\rho_{d1})) \tag{7}$$

Note that the correlations can be either positive or negative. The GD is reduced to a Dirichlet when $\beta_d = \alpha_{d+1} + \beta_{d+1}$ [14]. The GD is also a conjugate prior to the multinomial distribution. Thus, we can derive Multinomial Generalized Dirichlet distribution by integration over the multinomial parameters τ. If a random vector $X_i = [x_1 \cdots x_D]$, follows a Multinomial Generalized Dirichlet distribution [11], then we have:

$$MGD(X_i|\xi) = \frac{\Gamma(n+1)}{\prod_{d=1:x_d \geq 1}^{D+1} \Gamma(x_d + 1)} \prod_{d=1:x_d \geq 1}^{D} \frac{\Gamma(\alpha_d + \beta_d)}{\Gamma(\alpha_d)\Gamma(\beta_d)} \frac{\Gamma(\alpha_d^{'} + \beta_d^{'})}{\Gamma(\alpha_d^{'})\Gamma(\beta_d^{'})} \tag{8}$$

where $n = \sum_{d}^{D+1} x_d$, $\alpha_d^{'} = \alpha_d + x_d$, $\beta_d^{'} = \beta_d + x_{d+1} \cdots + x_{D+1}$, for $d = 1, \cdots, D$. The efficiency of MGD to model sparse high-dimensional count data can be improved by approximating it to belong to the exponential family. The authors in [60] found, experimentally, that $\alpha_d \ll \beta_d \ll 1$ for almost all words w based on different data sets. Moreover, for $x \geq 1$, we have [30]:

$$\lim_{\alpha \to 0} \frac{\Gamma(\alpha + x)}{\Gamma(\alpha)} - \alpha\Gamma(x) = 0 \tag{9}$$

Then, the exponential-family form for MGD can be written as [62]:

$$\begin{aligned} EMGD(X_i|\xi) \propto & \left(\prod_{D:x_D \geq 1} x_d^{-1} \right) \prod_{D:x_D \geq 1} \frac{\Gamma(z_d)}{\Gamma(x_d + z_d)} n \\ & \times \{ exp \sum_{d=1}^{D} I(x_d >= 1) log \frac{\alpha_d \beta_d}{\alpha_d + \beta_d} \} \end{aligned} \tag{10}$$

where $I(x_d >= 1)$ is an indicator that represents whether a word w shows up at any entry in the vector X_i, and $z_d = x_{d+1} + \cdots + x_{D+1}$.

1.2.8 The Exponential Family Approximation to Multinomial Beta-Liouville Distribution

The Beta-Liouville also is a conjugate prior to the multinomial distribution, and it has two parameters that can be used to adjust the spread of the distribution which make it more practical and provide better modeling capabilities, compared with Dirichlet distribution. If a random vector $X = (x_1, \cdots, x_{D+1})$, follows a Multinomial Beta-Liouville distribution, then [10]:

$$MBL(X|\xi) = \frac{\Gamma((\sum_{d=1}^{D+1} x_d) + 1)}{\prod_{d=1}^{D+1} \Gamma(x_d + 1)} \times \frac{\Gamma(\sum_{d=1}^{D} \alpha_d)\Gamma(\alpha + \beta)\Gamma(\alpha^{'})\Gamma(\beta^{'}) \prod_{d=1}^{D} \Gamma(\alpha_d^{'})}{\Gamma(\sum_{d=1}^{D} \alpha_d^{'})\Gamma(\alpha^{'} + \beta^{'})\Gamma(\alpha)\Gamma(\beta) \prod_{d=1}^{D} \Gamma(\alpha_d)} \tag{11}$$

where $\alpha_d^{'} = \alpha_d + x_d$, $\alpha^{'} = \alpha + \sum_{d=1}^{D} x_d$, $\beta^{'} = \beta + x_{D+1}$, and $\xi = (\alpha, \beta, \alpha_1, \ldots, \alpha_D)$.
In several real world applications, the MBL mixture model has provided good clustering accuracy, comparably to Multinomial Scaled Dirichlet mixture model (MSD) [59], and Multinomial Generalized Dirichlet mixture model (MGD) [9], it also outperforms other widely used mixture models, such as mixtures of Multinomial distributions (MM) and Dirichlet Compound Multinomial (DCM)

distributions [19,42]. Approximating MBL to belong to the exponential family can reduce the computation cost and improve the efficiency of MBL to model sparse high-dimensional count data [30]. The authors in [60] found empirically that $\alpha \ll 1$ and $\beta \simeq 1$ for real data sets with proposed maximum likelihood method for model parameters estimation. Thus, relying on Eq. 9, we have the form of exponential approximation for multinomial Beta-Liouville distribution as [60]:

$$EMBL(X|\xi) \propto (\prod_{d:x_d>=1} x_d^{-1})n!\frac{\Gamma(S)\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{\Gamma(S+n)\Gamma(\alpha^{'}+\beta^{'})} \times exp\{\sum_{d=1}^{D} I(x_d >= 1)log(\alpha_d)\} \quad (12)$$

where $I(x_d >= 1)$, the sufficient static, is an indicator whether the word d appears at least once in the vector X, and $S = \sum_{d=1}^{D} \alpha_d$.

1.2.9 The Exponential Family Approximation to Multinomial Scaled Dirichle Distribution

The Scaled Dirichlet is a generalization of the Dirichlet distribution obtained after applying the perturbation and powering operations to a Dirichlet random composition [59]. These operations define a vector-space structure in the simplex and play the same role as sum and product by scalars in real space. We assume the dimension is D, the scaled Dirichlet with a set of parameters $\alpha = (\alpha_1 \cdots \alpha_D)$ which is the shape parameter, and $\beta = (\beta_1 \cdots \beta_D)$ which is the scale parameter. Then, the Scaled Dirichlet distribution is defined by [1]:

$$SD(\rho|\alpha,\beta) = \frac{\Gamma(\alpha)\prod_{d=1}^{D}\beta_d^{\alpha_d}\rho_d^{\alpha_d-1}}{\prod_{d=1}^{D}\Gamma(\alpha_d)(\sum_{d=1}^{D}\beta_d\rho_d)^a} \quad (13)$$

where $a = \sum_{d=1}^{D} \alpha_d$. We note that the Scaled Dirichlet includes the Dirichlet as a special case when all elements of the vector β are equal to a common constant. Compared to the Dirichlet, the Scaled Dirichlet has D extra parameters, which enhances the model flexibility [34]. The good parameterization of scaled Dirichlet gives it the ability to model variance and covariance. Moreover, unlike Dirichlet, the scaled Dirichlet takes into account relative positions between categories or multinomial cells. These properties make the scaled Dirichlet a more flexible choice as a prior to the Multinomial.

The MSD model is composition of the Multinomial and scaled Dirichlet distribution, in this case, it has two parameters, which are shape parameter α and scale parameter β, and we assume $X_i = [x_1, \cdots, x_D]$. Thus, the MSD is defined by [59]:

$$\begin{aligned} MSD(X_i|\alpha,\beta) &= \int_\rho \mathcal{M}(X|\rho)\mathcal{SD}(\rho|\theta)d\rho \\ &= \frac{n!}{\prod_{d=1}^{D} x_d!}\frac{\Gamma(a)}{\Gamma(a+n)\prod_{d=1}^{D}\beta_d^{x_d}}\prod_{d=1}^{D}\frac{\Gamma(\alpha_d+x_d)}{\Gamma(\alpha_d)} \end{aligned} \tag{14}$$

where D is the vocabulary size, and $n = \sum_{d=1}^{D} x_d$.

The exponential family of distributions has obvious benefits such as simplicity, effective optimization, it retains the essential information in a dataset and reduces the computation time in high-dimensional data. Thus, the MSD ditribution has been approximated as a member of the exponential family in [59], using Eq. 9 for small α values. Thus, the $\mathcal{EMSD}$ distribution can be written as follows:

$$\mathcal{EMSD}(\mathbf{X_i}|\alpha,\beta) = \frac{n!\Gamma(S)}{\prod_{d=1,x_d\geq 1}^{D} x_d\Gamma(S+n)}\prod_{d=1,x_d\geq 1}^{D}\frac{\alpha_d}{\beta_d^{x_d}} \tag{15}$$

where $S = \sum_{d=1}^{D}\alpha_d$.

1.2.10 The Exponential Family Approximation to Multinomial Shifted-Scaled Distribution

In dimension D, the SSD distribution with parameters $\xi = \{\alpha, \beta, \tau\}$ is given by [61]:

$$SSD(\rho|\xi) = \frac{\Gamma(\alpha)}{\prod_{d=1}^{D}\Gamma(\alpha_d)}\frac{1}{\tau^{D-1}}\frac{\prod_{d=1}^{D}\beta_d^{-(\alpha_d/\tau)}\rho_d^{(\alpha_d/\tau)-1}}{\sum_{d=1}^{D}(\frac{p_d}{\beta_d})^{(1/\tau)^{\alpha'}}} \tag{16}$$

where $\alpha = (\alpha_1, \cdots, \alpha_D)$, $\beta = (\beta_1, \cdots, \beta_D)$, $\alpha' = \sum_{d=1}^{D}\alpha_d$, and τ is a constant. Compared with SD, SSD keeps (2D+1) degree of freedom that shows more flexible ability for real data applications. Note that SD is a special case for SSD, when $\tau = 1$. We can obtain multinomial shifted-scaled

Dirichlet distribution (MSSD) by integrating over multinomial parameters ρ. Thus, we have:

$$MSSD(X|\xi) = \int_{\rho} \mathcal{M}(X|\rho)\mathcal{SD}(\rho|\theta)d\rho$$
$$\frac{n!\Gamma(S)}{\prod_{d:x_d\geq 1}^{D} x_d!\Gamma(S+\tau n)} \prod_{d:x_d\geq 1}^{D} \frac{\Gamma(\alpha_d+\tau x_d)}{\beta_d^{x_d}\Gamma(\alpha_d)} \tag{17}$$

where $S = \sum_d^D \alpha_d$, $n = \sum_{d=1}^D x_d!$ and $\xi = (\alpha, \beta, \tau)$.

For high dimensional data, the authors in [61] found that the value of α parameters are really small which combined with some approximation gave the exponential Multinomial Shifted-Scaled Dirichlet (EMSSD) as:

$$EMSSD(X|\xi) \propto n!\frac{\Gamma(S)\tau^D}{\Gamma(S+\tau n)} \prod_{d:x_d\geq 1}^{D} \frac{\alpha_d}{\beta_d^{x_d}(x_d\times\tau)} \tag{18}$$

1.3 Contributions

In this thesis, we present clustering algorithms based on finite and infinite mixtures of EMGD, EMBL, and EMSSD from Bayesian viewpoint using Gibbs sampling within M-H steps. These distributions have already shown excellent performances in clustering real-world high-dimensional count data sets with deterministic approach. The key contributions of this thesis are as follows:

(1) Determination of conjugate priors to EMGD, EMBL, EMSD, and EMSSD by taking into account the fact that these distributions are members of the exponential family.

(2) Presenting MCMC algorithms based on Gibbs sampling and Metropolis-Hastings for the parameters estimation of finite mixture models.

(3) Extending finite mixture models of EMBL, EMGD, and EMSSD to the infinite case and proposing clustering algorithms based on MCMC and Dirichlet process for parameters estimation.

(4) Through challenging applications that concern text sentiment analysis, text fake news detection and human face gender recognition, we show that the proposed algorithms are efficient for clustering sparse high-dimensional count data.

1.4 Thesis Structure

In Chapter 2, we develop conjugate prior distributions for EMBL, EMGD, and EMSSD. Then, we present a Bayesian estimation for their finite mixture models parameters using Gibbs sampling, and extend finite mixture models of EMBL, EMGD, and EMSSD to infinite mixture models while developing complete clustering algorithms. Chapter 3 is devoted to exhibit the abilities of the proposed approaches in text sentiment analysis, text fake news detection, human face gender recognition. The concluding remarks and future work directions are given in Chapter 4.

Chapter 2

The Proposed Bayesian Learning Framework

In this chapter, we propose the algorithms to learn the parameters of finite and infinite mixture models of EMBL, EMGD, and EMSSD.

2.1 Bayesian Learning for Finite Mixture Weight Parameters

Given a set of N independent vectors $\mathcal{X} = \{X_1 \cdots X_N\}$ described by a finite mixture model, and M is the number of mixture components, we define an indicator for each X_i in data set $\mathcal{X}$ for each class j as:

$$Z_{ij} = \begin{cases} 1 & if\ X_i\ belongs\ to\ class\ j \\ 0 & otherwise \end{cases} \tag{19}$$

where $\mathbf{Z} = \{Z_1, \cdots, Z_N\}$ and $Z_i = (Z_{i1}, \cdots, Z_{iM})$.

In the Bayesian paradigm information brought by the complete data $(\mathcal{X}, \mathbf{Z})$, a realization of $(\mathcal{X}, \mathbf{Z}) \sim p(\mathcal{X}, \mathbf{Z}|\Theta)$ is combined with prior information about the parameters Θ that is specified in a prior distribution with density $\pi(\Theta)$ and summarized in probability distribution $\pi(\Theta|\mathcal{X}, \mathbf{Z})$ called the posterior distribution. This can be derived from the joint distribution, $p(\mathcal{X}, \mathbf{Z}|\Theta)\pi(\Theta)$ [13]. Thus,

we have:

$$\begin{aligned}\pi(\Theta|\mathcal{X},\mathbf{Z}) &= \frac{\pi(\Theta)p(\mathcal{X},\mathbf{Z}|\Theta)}{\int \pi(\Theta)p(\mathcal{X},\mathbf{Z}|\Theta)} \\ &\propto \pi(\Theta)\times p(\mathcal{X},\mathbf{Z}|\Theta)\end{aligned} \tag{20}$$

where $\int \pi(\Theta)p(\mathcal{X},\mathbf{Z}|\Theta)$ is the marginal density of the complete data $(\mathcal{X},\mathbf{Z})$.

We can directly simulate $\Theta \sim \pi(\Theta|\mathcal{X},\mathbf{Z})$ with well-known Gibbs sampler. The Gibbs sampling is widely used in Bayesian mixture model, especially in the case of incomplete data [50, 52]. That is associated with each observation X_i a missing multinomial variables $\mathbf{Z} \sim \mathbf{M}(1, Z_{i1}\cdots Z_{iM})$.

$$Z_{ij} = \frac{p(X_j|\xi_j)P_j}{\sum_{j=1}^{M} p(X_j|\xi_j)P_j} \tag{21}$$

In fact, the weight parameters is independent of $\mathcal{X}$, $P_j \propto \pi(P|\mathbf{Z})$ [40], and we know that the vector P is defined on the simplex $\{(P_1,\cdots,P_M); \sum_{j=1}^{M-1} P_j < 1\}$, then the natural prior distribution for vector P is the Dirichlet distribution, we determine the prior $\pi(P)$ [7] as:

$$\pi(P|\eta_j) = \frac{\Gamma(\sum_{j=1}^{M}\eta_j)}{\prod_{j=1}^{M}\Gamma(\eta_j)}\prod_{j=1}^{M} P_j^{\eta_j-1} \tag{22}$$

where $\eta = (\eta_1,\cdots,\eta_M)$ is the parameters vector of the Dirichlet distribution. Moreover, we have:

$$\begin{aligned}\pi(\mathbf{Z}|P) &= \prod_{i=1}^{N}\pi(Z_i|P) = \prod_{i=1}^{N} P_1^{Z_{i1}}\cdots P_M^{Z_{iM}} \\ &= \prod_{i=1}^{N}\prod_{j=1}^{M} P_j^{Z_{ij}} = \prod_{j=1}^{M} P_j^{n_j}\end{aligned} \tag{23}$$

where $n_j = \sum_{i=1}^{N} I_{Z_{ij}=1}$.

Having the prior distribution and likelihood distribution in hand, we can obtain the posterior for

weight parameter P by the following:

$$\begin{aligned}\pi(P|\mathbf{Z}) &\propto \pi(P)\pi(\mathbf{Z}|P) \\ &= \prod_{j=1}^{M} P_j^{n_j} \frac{\Gamma(\sum_{j=1}^{M}\eta_j)}{\prod_{j=1}^{M}\Gamma(\eta_j)} \prod_{j=1}^{M} P_j^{\eta_j - 1} \\ &= \frac{\Gamma(\sum_{j=1}^{M}\eta_j)}{\prod_{j=1}^{M}\Gamma(\eta_j)} \prod_{j=1}^{M} P_j^{\eta_j + n_j - 1} \\ &\propto D(\eta_1 + n_1 \cdots \eta_M + n_M)\end{aligned} \tag{24}$$

where D is Dirichlet distribution with parameters $(\eta_1 + n_1, \cdots, \eta_M + n_M)$. We note that the prior and posterior distributions $\pi(P)$ and $\pi(P|\mathbf{Z})$ are both Dirichlet distributions, In this case, we say that the Dirichlet is a conjugate prior for mixture proportions. Therefore, the weight parameters can be sampled from Dirichlet distribution.

2.2 Bayesian Learning for Infinite Mixture Weight Parameters

In finite mixture model, we have considered M to be fixed finite quantity. In this section, we will explore the limit $M \to \infty$ and present the conditional posteriors for the indicators and weight parameters based on Dirichlet process. We take $(\eta_1, \cdots, \eta_M) = (\eta/M, \cdots, \eta/M)$ for Eq. 22, thus we obtain a simpler form for prior probability of infinite mixture weight parameters [49]:

$$\pi(P_{inf}|\eta) = \frac{\Gamma(\eta)}{\Gamma(\eta/M)^M} \prod_{j=1}^{M} \pi_j^{\eta/M - 1} \tag{25}$$

where we have $P_{inf} = (P_{inf^1}, \cdots, P_{inf^M})$. From Eq. 23, we have the prior distribution for the Z parameter that corresponds to multinomial distribution. Using the standard Dirichlet integral, we could marginalize out the P_{inf} parameter to get the following probability for the prior directly in

terms of the indicators [47]:

$$\begin{aligned} p(\mathbf{Z}|\eta) &= \int P(\mathbf{Z}|P_{inf})P(P_{inf}|\eta) \\ &= \frac{\Gamma(\eta)}{\Gamma(\eta/M)^M} \int \prod_{j=1}^{M} \pi^{n_j+\eta/M-1} d\pi_j \\ &= \frac{\Gamma(\eta)}{\Gamma(N+\eta)} \prod_{j=1}^{M} \frac{\Gamma(n_j+\eta/M)}{\Gamma(\eta/M)} \end{aligned} \tag{26}$$

Based on Bayes principle, we obtain the conditional posterior distribution for the mixing weight vector:

$$\begin{aligned} \pi(P_{inf}|\mathbf{Z}) &= \frac{p(P_{inf}|\eta)p(\mathbf{Z}|P_{inf})}{p(\mathbf{Z}|\eta)} \\ &= \prod_{j=1}^{M} P_{inf^j}^{n_j} \frac{\Gamma(\sum_{j=1}^{M}\eta)}{\prod_{j=1}^{M}\Gamma(\eta_j)} \prod_{j=1}^{M} P_{inf^j}^{\eta/M-1} \\ &= \frac{\Gamma(\sum_{j=1}^{M}\eta/M)}{\prod_{j=1}^{M}\Gamma(\eta/M)} \prod_{j=1}^{M} P_{inf^j}^{\eta/M+n_j-1} \\ &\propto D(\eta/M+n_1 \cdots \eta/M+n_M) \end{aligned} \tag{27}$$

In order to be able to use Gibbs sampling for the indicators Z_i, we need the conditional prior for a single indicator given all the others: this is easily obtained from Eq. 26 by keeping all but a single indicator fixed [46]:

$$p(Z_i = j|Z\_i, \eta) = \frac{n_{\_i,j} + \eta/M}{N - 1 + \eta} \tag{28}$$

where the subscript $-i$ indicates all except i and $n_{\_i,j}$ is the number of observations, excluding X_i, that are associated with component j.

Lastly, we choose inverse Gamma as prior for parameters η:

$$p(\eta|\vartheta, \varrho) = \frac{\varrho^{\vartheta} exp(-\varrho/\eta)}{\Gamma(\vartheta)\eta^{\vartheta+1}} \tag{29}$$

The likelihood for η can be derived from Eq. 26, which together with the prior from Eq. 29 gives:

$$p(\eta|\vartheta, \varrho, M, N) = \frac{\varrho^{\vartheta} exp(-\varrho/\eta)}{\Gamma(\vartheta)\eta^{\vartheta+1}} \times \frac{\eta^M \Gamma(\eta)}{\Gamma(N+\eta)} \tag{30}$$

For the indicators, letting $M \rightarrow \infty$ in Eq. 28, the conditional prior reaches the following limits [47]:

$$p(Z_i = j|\eta, Z_{\_i}) = \begin{cases} \frac{n_{\_i,j}}{N-1+\eta} & if\ n_{\_ij} > 0 \\ \frac{\eta}{N-1+\eta} & if\ n_{\_ij} = 0 \end{cases} \tag{31}$$

Having this prior distribution, we can obtain the conditional posterior by multiplying the model likelihood:

$$p(Z_i = j|\eta, Z_{\_i}) = \begin{cases} \frac{n_{\_i,j}}{N-1+\eta} p(\mathcal{X}|\xi) & if\ n_{\_ij} > 0 \\ \int \frac{\eta}{N-1+\eta} p(\mathcal{X}|\xi)p(\xi)d\xi & if\ n_{\_ij} = 0 \end{cases} \tag{32}$$

Unfortunately, this integral is not analytically tractable in Eq. 32, hence, we consider a Monte Carlo sampling approximation.

2.3 Learning Algorithms for Finite and Infinite Models

In this section, we propose the algorithms to learn the parameters for finite and infinite mixture models of EMBL, EMGD and EMSSD.

2.3.1 Learning Algorithm for Finite Mixture Model of EMGD

Define $\pi(\xi)$ as the prior distribution for the parameters of the EMGD distribution. We use the fact that EMGD belongs to the exponential family. In fact, if a S-parameters density ρ belong to the exponential family then we can rewrite it in the exponential form [32] which has been shown in Eq. 4. Writing the EMGD in the exponential form gives:

$$\begin{aligned} H(X) &= (\prod_{W:x_W \geq 1} x_w^{-1}) \prod_{W:x_W \geq 1} \frac{\Gamma(z_w)}{\Gamma(x_w + z_w)} n! \\ G_w(\xi) &= log \frac{\alpha_w \beta_w}{\alpha_w + \beta_w} \\ T_w(X) &= \sum_{w=1}^{W} I(x_w >= 1) \\ \phi(\xi) &= 0 \end{aligned} \tag{33}$$

In this case, a prior of ξ is given by [7] as:

$$\pi(\xi) \propto exp(\sum_{w=1}^{W} \rho_l G_w(\xi) + k\Phi(\xi)) \tag{34}$$

where $\rho = (\rho_1, \cdots, \rho_w)$, and $k > 0$ are referred as hyperparameters.

The prior for EMGD can be written as follows:

$$\pi(\alpha, \beta) \propto exp(\sum_{w=1}^{W} \rho_l log \frac{\alpha_w \beta_w}{\alpha_w + \beta_w}) \tag{35}$$

Having the prior in hand, the mixture model posterior is (see Appendix. A.1):

$$\begin{aligned} \pi(\xi_j|\mathbf{M}, \mathbf{X}) &\propto \pi(\xi_j) \prod_{Z_{ij}=1} EMGD(X_i|\xi_j) \\ &\propto exp[\sum_{w=1}^{W} log \frac{\alpha_w \beta_w}{\alpha_w + \beta_w} (\rho_w + \sum_{Z_{ij}=1}^{N} I(x_{iw} \geq 1))] \\ &\times \prod_{Z_{ij}=1}^{N} (\prod_{w:x_{iw} \geq 1} x_{iw}^{-1} \frac{\Gamma(z_{iw})}{\Gamma(x_{iw} + z_{iw})} n!) \end{aligned} \tag{36}$$

According to the posterior hyperparameters, following [5, 13], once the sample $\mathbf{X}$ is known, we can use it to get the prior hyperparameters. Then, we held $(\rho_1, \cdots, \rho_W)$ and $(\eta_1, \cdots, \eta_M)$ fixed at: $\eta_j = 1, j = 1 \cdots M$, $\rho_w = 1, w = 1 \cdots W$.

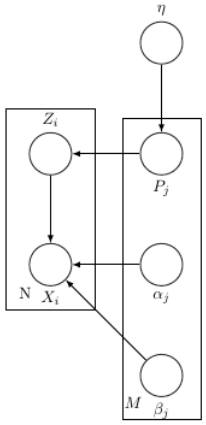

Figure 2.1: Graphical Model Representation of the Finite EMGD

Algorithm 3 Finite EMGD (FinEMGD) learning algorithm

Initialization: Using MOM and K-means method to initialize model parameters and cluster number selection

Input: A data set $\mathcal{X} = \{X_1 \cdots X_N\}$, each is W-dimensional sparse count vector

output: Θ

for $t = 1 \cdots$:

(1) Generate $\mathbf{Z^t} \sim \mathcal{M}(1; \hat{Z_{i1}}^{t-1} \cdots \hat{Z_{iM}}^{t-1})$

(2) Generate weight parameters P^t from Eq. 27

(3) Generate model ξ^t from Eq. 36 using Metropolis-Hasting(M-H) algorithm

Metropolis-Hasting(M-H) algorithm:

(1) Generate $\widetilde{\xi}_j$ from $q(\widetilde{\xi}_j|\xi_j^{t-1})$ and $u \sim U[0,1]$

(2) compute $r = \frac{\pi(\widetilde{\xi}_j|\mathbf{M},\mathbf{X})q(\xi_j^{t-1}|\widetilde{\xi}_j)}{\pi(\xi_j^{t-1}|\mathbf{M},\mathbf{X})q(\widetilde{\xi}_j|\xi_j^{t-1})}$

(3) if $r < u$ then: $\xi^t = \widetilde{\xi}$ else: $\xi^t = \xi^{t-1}$

In Algorithm 3, $\xi_j = (\alpha_{j1}, \beta_{j1}, \cdots, \alpha_{jW}, \beta_{jW})$, and we take the K-means [58] and the method of moments (MOM) [57] for initializing the model parameters. In the Metropolis-Hastings (M-H) step, the major factor is choosing proposal distribution q [23, 52]. As the model parameters are satisfied $0 < \alpha_{jw} \ll \beta_{jw} \leq 1$, we choose the Gamma distribution as the proposal distribution for α_{jw} and β_{jw}.

$$\alpha_{jw} \sim \mathcal{G}(\alpha, \sigma_1), \beta_{jw} \sim \mathcal{G}(\beta, \sigma_2) \tag{37}$$

where σ_1 and σ_2 are scale parameters of the Gamma distributions.

The complexity of the algorithm is determined by the size of data set (i.e., number of observations N), and the number of mixture components K. The algorithm computation complexity for one iteration is $O(NK)$. The complete algorithm for estimating the EMGD parameters using the proposed approach is presented in Algorithm 3.

2.3.2 Learning Algorithm for Infinite Mixture Model of EMGD

We know that the model parameters α and β in EMGD satisfy $0 < \alpha_{jw} \ll \beta_{jw} < 1$, then appealing flexible choice as prior is the Beta distribution, with shape parameters: δ, ϵ and ϖ, ϱ,

then:

$$p(\alpha_j) \propto \frac{\Gamma(\delta+\epsilon)}{\Gamma(\delta)\Gamma(\epsilon)}\alpha_j^{\delta-1}(1-\alpha_j)^{\epsilon-1} \tag{38}$$

$$p(\beta_j) \propto \frac{\Gamma(\varpi+\varrho)}{\Gamma(\varpi)\Gamma(\varrho)}\beta_j^{\varpi-1}(1-\beta_j)^{\varrho-1} \tag{39}$$

where $\alpha_j = (\alpha_{j1}, \cdots, \alpha_{jD})$, $\beta_j = (\beta_{j1}, \cdots \beta_{jD})$.

Then, the conditional posterior distributions for α_j and β_j are:

$$\begin{aligned} p(\alpha_j|\mathcal{X}, \mathbf{Z}) \propto p(\alpha_j) & \prod_{Z_{ij}=1} EGDM(X_i|\xi_j) \\ & \frac{\Gamma(\delta+\epsilon)}{\Gamma(\delta)\Gamma(\epsilon)}\alpha_j^{\delta-1}(1-\alpha_j)^{\epsilon-1} \prod_{Z_{ij}=1} \{(\prod_{W:x_W\geq 1} x_w^{-1}) \prod_{W:x_W\geq 1} \frac{\Gamma(z_w)}{\Gamma(x_w+z_w)} n \\ & \times \{exp \sum_{w=1}^{W} I(x_w >= 1) log \frac{\alpha_w \beta_w}{\alpha_w+\beta_w}\}\} \end{aligned} \tag{40}$$

$$\begin{aligned} p(\beta_j|\mathcal{X}, \mathbf{Z}) \propto p(\beta_j) & \prod_{Z_{ij}=1} EGDM(X_i|\xi_j) \\ & \frac{\Gamma(\varpi+\varrho)}{\Gamma(\varpi)\Gamma(\varrho)}\beta_j^{\varpi-1}(1-\beta_j)^{\varrho-1} \prod_{Z_{ij}=1} \{(\prod_{W:x_W\geq 1} x_w^{-1}) \prod_{W:x_W\geq 1} \frac{\Gamma(z_w)}{\Gamma(x_w+z_w)} n \\ & \times \{exp \sum_{w=1}^{W} I(x_w >= 1) log \frac{\alpha_w \beta_w}{\alpha_w+\beta_w}\}\} \end{aligned} \tag{41}$$

In order to have more flexible model, we introduce an additional hierarchical level by allowing the hyperparmeters to follow some selected distributions. The hyperparmeters δ, ϵ and ϖ, ϱ associated with α and β respectively are given Beta distribution and Exponential distribution:

$$p(\delta|\varsigma, \upsilon) = \frac{\Gamma(\varsigma+\upsilon)}{\Gamma(\varsigma)\Gamma(\upsilon)}\delta^{\varsigma-1}(1-\varsigma)^{\upsilon-1} \tag{42}$$

$$p(\epsilon|\lambda) = \lambda exp(-\lambda\epsilon) \tag{43}$$

$$p(\varpi|\kappa, \omega) = \frac{\Gamma(\kappa+\omega)}{\Gamma(\kappa)\Gamma(\omega\varpi^{\kappa-1}(1-\kappa)^{\omega-1}} \tag{44}$$

$$p(\varrho|\iota) = \iota exp(-\iota\varrho) \tag{45}$$

For those hyperparameters δ, ϵ and ϖ, ϱ, the prior of α and β is considered as likelihood. Thus, the conditional posterior can be obtained (see Appendix A.4).

Then, we have the learning Algorithm 4 for infinite mixture model of EMGD:

Algorithm 4 Infinite EMGD (InfEMGD) learning algorithm

Initialization: Using MOM to initialize model parameters
Input: A data set $\mathcal{X} = \{X_1 \cdots X_N\}$, each is W-dimensional sparse count data
output: Θ
for $t = 1 \cdots$:

(1) Generate $\mathbf{Z^t}$ from Eq. 31 with Monte Carlo sampling approximation

(2) Update the number of represented components

(3) Generate parameters η from Eq. 27 with adaptive reject sampling (ARS)

(4) Generate weight parameters P^t from $Dir(\eta/M + n_1, \cdots, \eta/M + n_M)$

(5) Update α, β in Metropolis-Hasting(M-H) algorithm

Metropolis-Hasting(M-H) algorithm:
for γ_j in (α_j, β_j) :

(1) Generate $\widetilde{\gamma}_j$ from $q(\widetilde{\gamma}_j|\gamma_j^{t-1})$ and $u \sim U[0,1]$

(2) compute $r = \frac{p(\widetilde{\gamma}_j|\mathbf{M},\mathbf{X})q(\gamma_j^{t-1}|\widetilde{\gamma}_j)}{p(\gamma_j^{t-1}|\mathbf{M},\mathbf{X})q(\widetilde{\gamma}_j|\gamma_j^{t-1})}$ from Eq. (39) or Eq. (40)

(3) if $r < u$ then: $\xi^t = \widetilde{\xi}$ else: $\xi^t = \xi^{t-1}$

Update the hyperparameters δ, ϵ and ϖ, ϱ with MCMC sampling in their conditional posterior

2.3.3 Learning Algorithm for Finite Mixture Model of EMBL

EMBL also belongs to the exponential family. We define $\mathcal{X} = \{X_1, \cdots, X_N\}$, where $X_i = [x_{i1} \cdots x_{iW}]$. We can show following 4 [32], that:

$$\begin{aligned} H(X) &= (\prod_{W:x_W \geq 1} x_w^{-1})n! \\ G_w(\xi) &= log(\alpha_w) \\ T_w(X) &= \sum_{w=1}^{W} I(x_w >= 1) \\ \phi(\xi) &= log\{\frac{\Gamma(\alpha^{'})\Gamma(s)\Gamma(\beta^{'})\Gamma(\alpha)\alpha}{\Gamma(s+n)\Gamma(\alpha^{'}+\beta^{'})}\} \end{aligned} \tag{46}$$

Thus, we have a prior as follows:

$$\pi(\alpha, \beta) \propto exp[\sum_{w=1}^{W} \rho_d log(\alpha_d) + k(log(\frac{\Gamma(s)\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{\Gamma(s+n)\Gamma(\alpha^{'}+\beta^{'})}))] \tag{47}$$

From Bayesian theory, the posterior can be written as (see Appendix A.2):

$$\begin{aligned} \pi(\xi_j|\mathbf{M}, \mathbf{X}) &\propto \pi(\xi_j) \prod_{Z_{ij}=1} EMBL(X_i|\xi_j) \\ &\propto exp[\sum_{w=1}^{W} log(\alpha_w)(\rho_w + \sum_{Z_{ij}=1}^{N} I(x_{iw} \geq 1)) \\ &+ k(log(\frac{\Gamma(\alpha_j^{'})\Gamma(\beta_j^{'})\alpha_j}{(S) \times (S+1) \cdots (S+n-1)\Gamma(\alpha_j^{'}+\beta_j^{'})})) \\ &+ \sum_{i=1, z_{ij}=1} (log(\frac{\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{(S) \times (S+1) \cdots (S+n-1)\Gamma(\alpha^{'}+\beta^{'})}))] \end{aligned} \tag{48}$$

Once the sample $\mathcal{X}$ is known, the posterior hyperparameters can be fixed, we fix $\rho_w = 1$, $k = 1$ and $\eta = 1$ [5, 13].

In Bayesian approach, choosing an effective proposal prior distribution is significant factor for the model parameters estimation and convergence time. With many different common proposal

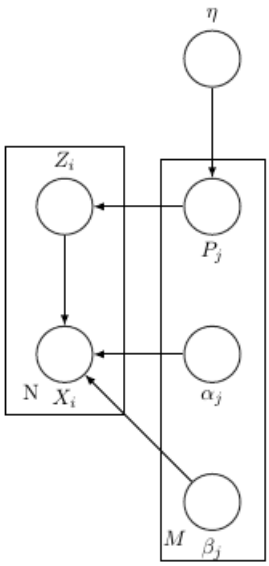

Figure 2.2: Graphical Model Representation of the Finite EMBL.

distributions attempts, we finally select Beta distribution as proposal distribution for α_{jw}, and inv-Gamma distribution for β:

$$\alpha_{jw} \sim \mathcal{B}(\alpha, \sigma_1), \beta \sim invG(\beta, \sigma_2) \tag{49}$$

The complete steps for estimating the EMBL model parameters using the proposed approach are given in Algorithm 5. Note that the proposed Algorithm 5 requires computational cost $O(NK)$ per step.

2.3.4 Learning Algorithm for Infinite Mixture Model of EMBL

As shown empirically, the values of α and β satisfy $0 < \alpha \ll 1$ and $\beta \simeq 1$. Thus, we choose the Beta distribution and Inverse Gamma distribution as priors for α and β with hyperparameters δ, ϵ and ϖ, ϱ, then

$$p(\alpha_j) \sim \frac{\Gamma(\delta+\epsilon)}{\Gamma(\delta)\Gamma(\epsilon)} \alpha_j^{\delta-1}(1-\alpha_j)^{\epsilon-1} \tag{50}$$

$$p(\beta_j) \sim \frac{\varrho^{\varpi} exp(-\varrho/\beta_j)}{\Gamma(\varpi)\beta_j^{\varpi-1}} \tag{51}$$

Algorithm 5 Finite EMBL (FinEMBL) learning algorithm

Initialization: Using the MOM and the K-means method to initialize model parameters
Input: A data set $\mathcal{X} = \{X_1 \cdots X_N\}$, each is W-dimensional sparse count data
output: Θ
for $t = 1 \cdots$:

(1) Generate $\mathbf{Z^t} \sim \mathcal{M}(1; \hat{Z_{i1}}^{t-1} \cdots \hat{Z_{iM}}^{t-1})$

(2) Generate weight parameters P^t from Eq. 24

(3) Generate model ξ^t from Eq. 48 using Metropolis-Hasting(M-H) algorithm

Metropolis-Hasting(M-H) algorithm:

(1) Generate $\widetilde{\xi}_j$ from $q(\widetilde{\xi}_j|\xi_j^{t-1})$ and $u \sim U[0,1]$

(2) compute $r = \frac{\pi(\widetilde{\xi}_j|\mathbf{M},\mathbf{X})q(\xi_j^{t-1}|\widetilde{\xi}_j)}{\pi(\xi_j^{t-1}|\mathbf{M},\mathbf{X})q(\widetilde{\xi}_j|\xi_j^{t-1})}$

(3) if $r < u$ then: $\xi^t = \widetilde{\xi}$ else: $\xi^t = \xi^{t-1}$

Having this prior, the full conditional posteriors for α_j and β_j are:

$$\begin{aligned} p(\alpha_j|\mathcal{X},\mathbf{Z}) &\propto p(\alpha_j) \prod_{Z_{ij}=1} P_{EMBL}(X_i|\xi_j) \\ &\propto \frac{\Gamma(\delta+\epsilon)}{\Gamma(\delta)\Gamma(\epsilon)} \alpha_j^{\delta-1}(1-\alpha_j)^{\epsilon-1} \prod_{Z_{ij}=1} \{(\prod_{d:x_d>=1} x_d^{-1}) n! \\ &\propto \frac{\Gamma(S)\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{\Gamma(S+n)\Gamma(\alpha^{'}+\beta^{'})} \times exp\{\sum_{d=1}^{D} I(x_d \geq 1) log(\alpha_d)\}\} \end{aligned} \tag{52}$$

$$\begin{aligned} p(\beta_j|\mathcal{X},\mathbf{Z}) &\propto p(\beta_j) \prod_{Z_{ij}=1} P_{EMBL}(X_i|\xi_j) \\ &\propto \frac{\varrho^{\varpi} exp(-\varrho/\beta_j)}{\Gamma(\varpi)\beta_j^{\varpi-1}} \prod_{Z_{ij}=1} \{(\prod_{d:x_d>=1} x_d^{-1}) n! \\ &\propto \frac{\Gamma(S)\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{\Gamma(S+n)\Gamma(\alpha^{'}+\beta^{'})} \times exp\{\sum_{d=1}^{D} I(x_d \geq 1) log(\alpha_d)\}\} \end{aligned} \tag{53}$$

In order to reduce the sensitivity of parameters, we give priors for the hyperparamete δ, ϵ and ϖ, ϱ, by choosing Beta distribution, exponential distribution and Inverse Gamma distribution, exponential distribution, respectively.

$$p(\delta|\varsigma,\upsilon) \sim \frac{\Gamma(\varsigma+\upsilon)}{\Gamma(\varsigma)\Gamma(\upsilon)} \delta^{\varsigma-1}(1-\varsigma)^{\upsilon-1} \tag{54}$$

$$p(\epsilon|\lambda) \sim \lambda exp(-\lambda\epsilon) \tag{55}$$

$$p(\varpi|\kappa,\omega) \sim \frac{\omega^{\kappa} exp(-\omega/\varpi)}{\Gamma(\kappa)\varpi_j^{\kappa-1}} \tag{56}$$

$$p(\varrho|\iota) \sim \iota exp(-\iota\varrho) \tag{57}$$

For those hyperparameters δ, ϵ and ϖ, ϱ, the prior of α and β is considered as likelihood. Thus, the conditional posterior can be obtained (see Appendix A.4).

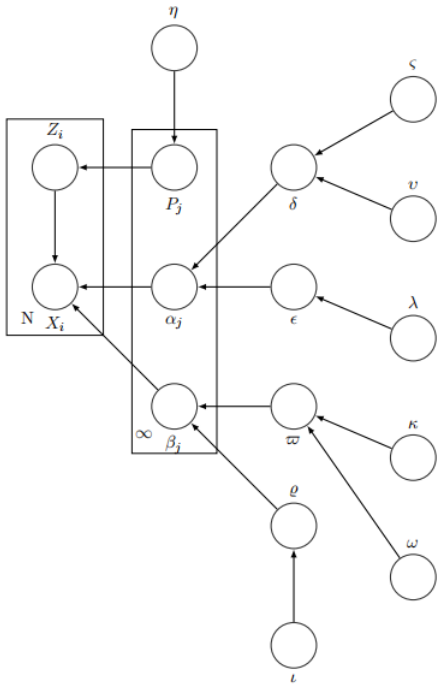

Figure 2.3: Graphical Model Representation of the Infinite EMGD or EMBL.

The parameter learning algorithm of this infinite model is similar to that infinite mixture model of EGDM, we only need to replace the posterior probability for α, β and $\delta, \epsilon, \varpi, \varrho$ in M-H steps. Thus, we have the learning Algorithm 6:

Algorithm 6 Infinite EMBL (InfEMBL) learning algorithm

Initialization: Using MOM to initialize model parameters

Input: A data set $\mathcal{X} = \{X_1 \cdots X_N\}$, each is W-dimensional sparse count data

output: Θ

for $t = 1 \cdots$:

(1) Generate $\mathbf{Z^t}$ from Eq. 31 with Monte Carlo sampling approximation

(2) Update the number of represented components

(3) Generate parameters η from Eq. 27 with adaptive reject sampling (ARS)

(4) Generate weight parameters P^t from $Dir(\eta/M + n_1, \cdots, \eta/M + n_M)$

(5) Update α, β in Metropolis-Hasting(M-H) algorithm

Metropolis-Hasting(M-H) algorithm:

for γ_j in (α_j, β_j) :

(1) Generate $\widetilde{\gamma}_j$ from $q(\widetilde{\gamma}_j|\gamma_j^{t-1})$ and $u \sim U[0, 1]$

(2) compute $r = \frac{p(\widetilde{\gamma}_j|\mathbf{M},\mathbf{X})q(\gamma_j^{t-1}|\widetilde{\gamma}_j)}{p(\gamma_j^{t-1}|\mathbf{M},\mathbf{X})q(\widetilde{\gamma}_j|\gamma_j^{t-1})}$ from Eq. (43) or Eq. (44)

(3) if $r < u$ then: $\xi^t = \widetilde{\xi}$ else: $\xi^t = \xi^{t-1}$

Update the hyperparameters δ, ϵ and ϖ, ϱ with MCMC sampling in their conditional posterior

2.3.5 Learning Algorithm for Finite Mixture Model of EMSD

The exponential family also includes EMSD. We can show following Eq. 4, that:

$$
\begin{aligned}
H(\xi) &= n!(\prod_{x_d \geq 1} x_d^{-1}) \\
G_l(\xi) &= [\log(\alpha_{jd}) - \log(\beta_{jd})] \\
T_l(x) &= \begin{bmatrix} \sum_{d=1}^{D} I(x_d \geq 1) \\ \sum_{d=1}^{D} x_d \end{bmatrix} \\
\log(\Gamma(s+n_i)) &= \log(\Gamma(s)) + \sum_{t=1}^{n-1} \log(s+t) \\
\Phi_l(\xi) &= \log(\frac{\Gamma(s)}{\Gamma(s+n_i)}) = -\sum_{t=1}^{n-1} \log(s+t)
\end{aligned}
\tag{58}
$$

Thus, the EMSD can be rewritten as:

$$
\begin{aligned}
\mathcal{EMSD}(\mathbf{X_i}|\alpha_j, \beta_j) &= (\prod_{x_d \geq 1} {x_d}^{-1}) n! \frac{\Gamma(s)}{\Gamma(s+n)} \\
&= \left\{ \exp(\sum_{d=1}^{D} I(x_d \geq 1)(\log(\alpha_{jd}) - x_w \log(\alpha_{jd})) \right\}
\end{aligned}
\tag{59}
$$

where the $\xi_j = (\alpha_{jd}, \beta_{jd})$.

In this case, a conjugate prior for ξ is given by:

$$
\begin{aligned}
P(\xi_j) &\propto \exp(\sum_{d=1}^{D} \rho_l G_l(\xi_j) + k\Phi(\xi_j)) \\
&\propto \exp\Bigg[\sum_{d=1}^{D} (\rho_1 \log(\alpha_{jd}) - \\
&\qquad \rho_2 \log(\beta_{jd})) - k \sum_{t=1}^{n-1} \log(s+t) \Bigg]
\end{aligned}
\tag{60}
$$

Algorithm 7 Finite EMSD (FinEMSD) learning algorithm

Initialization: Using the MOM and the K-means method to initialize model parameters
Input: A data set $\mathcal{X} = \{X_1 \cdots X_N\}$, each is D-dimensional sparse count data
output: Θ
for $t = 1 \cdots$:

(1) Generate $\mathbf{Z^t} \sim \mathcal{M}(1; \hat{Z_{i1}}^{t-1} \cdots \hat{Z_{iM}}^{t-1})$

(2) Generate weight parameters P^t from Eq. 24

(3) Generate model ξ^t from Eq. 61 using Metropolis-Hasting(M-H) algorithm

Metropolis-Hasting(M-H) algorithm:

(1) Generate $\widetilde{\xi}_j$ from $q(\widetilde{\xi}_j|\xi_j^{t-1})$ and $u \sim U[0,1]$

(2) compute $r = \frac{\pi(\widetilde{\xi}_j|\mathbf{M},\mathbf{X})q(\xi_j^{t-1}|\widetilde{\xi}_j)}{\pi(\xi_j^{t-1}|\mathbf{M},\mathbf{X})q(\widetilde{\xi}_j|\xi_j^{t-1})}$

(3) if $r < u$ then: $\xi^t = \widetilde{\xi}$ else: $\xi^t = \xi^{t-1}$

where (ρ_1, ρ_2, k) are the prior's hyperparameters. Thus, we can determine the posterior distribution as follows:

$$
\begin{aligned}
P(\xi_j|\mathcal{X},\mathcal{Z}) &\propto P(\xi_j) EMSD(\mathbf{X_i}|\xi_j) \\
&\propto \exp\left\{\left\{\left[\sum_{d=1}^{D}(\log(\alpha_{jd})(\rho_1 + \sum_{i,z_{ij}=1} I(x_{id} \geq 1)) - \log(\beta_{jd})(\rho_2 + \sum_{i=1,z_{ij}=1} x_{id})\right]\right\}\right. \\
&\left. - k\sum_{t=1}^{n-1}\log(s+t) + \sum_{i,z_{ij}=1}\left\{\left[log(n!) - \sum_{d,x_{id}\geq 1} log(x_{id}) - \sum_{t=1}^{n_i-1}\log(s+t)\right]\right\}\right\}
\end{aligned}
\tag{61}
$$

Considering [5], once the sample $\mathcal{X}$ is known, it can be used to get the prior hyperparameters. The hyperparameters are fixed at: $k = 1$, $\rho_1 = 1$, $\rho_2 = 1, \eta = 1$.

2.3.6 Learning Algorithm for Finite Mixture Model of EMSSD

EMSSD can be written following Eq. 4, as:

$$
\begin{aligned}
H(X) &= \frac{n!}{\prod_{w:x_w \geq 1}^{D} x_i w} \\
G_{w1}(\xi) &= log(\alpha_w) - log(\tau) \\
G_{w2}(\xi) &= log(\beta_w) \\
T_{w1}(X) &= \sum_{w=1}^{W} I(x_w >= 1) \\
T_{w2}(X) &= \sum_{w=1}^{W} I(x_w >= 1) x_w \\
\phi(\xi) &= log\{\frac{\Gamma(\alpha_+)\tau^D}{\Gamma(\alpha_+ + \tau n)}\}
\end{aligned}
\tag{62}
$$

Thus, we have a prior as follows:

$$
\begin{aligned}
\pi(\xi) \propto exp[&\sum_{w=1}^{W}\{\rho_{1w}(log(\alpha_w) - log(\tau)) + \rho_{2w} log(\beta_w)\} \\
&+ k \times log\{\frac{\Gamma(\alpha_+)\tau^D}{\Gamma(\alpha_+ + \tau n)}\}
\end{aligned}
\tag{63}
$$

Algorithm 8 Finite EMSSD (FinEMSSD) learning algorithm

Initialization: Using MOM and K-means method to initialize model parameters
Input: A data set $\mathcal{X} = \{X_1 \cdots X_N\}$, each is W-dimensional sparse count data
output: Θ
for $t = 1 \cdots$:

(1) Generate $\mathbf{Z^t} \sim \mathcal{M}(1; \hat{Z_{i1}}^{t-1} \cdots \hat{Z_{iM}}^{t-1})$

(2) Generate weight parameters P^t from Eq. 24

(3) Generate model ξ^t from Eq. 64 using Metropolis-Hasting(M-H) algorithm

Metropolis-Hasting(M-H) algorithm:

(1) Generate $\widetilde{\xi}_j$ from $q(\widetilde{\xi}_j|\xi_j^{t-1})$ and $u \sim U[0,1]$

(2) compute $r = \frac{\pi(\widetilde{\xi}_j|\mathbf{M},\mathbf{X})q(\xi_j^{t-1}|\widetilde{\xi}_j)}{\pi(\xi_j^{t-1}|\mathbf{M},\mathbf{X})q(\widetilde{\xi}_j|\xi_j^{t-1})}$

(3) if $r < u$ then: $\xi^t = \widetilde{\xi}$ else: $\xi^t = \xi^{t-1}$

From Bayesian theory, the posterior can be written as

$$
\begin{aligned}
p(\xi_j|\mathcal{X},\mathbf{Z}) &\propto \pi(\xi_j) \prod_{Z_{ij}=1}^{N} EMSSD(X|\xi_j) \\
&= exp[\sum_{w=1}^{W}\{\rho_{1w}(log(\alpha_w) - log(\tau)) + \rho_{2w} log(\beta_w)\} \\
&\quad + k \times log\{\frac{\Gamma(\alpha_+)\tau^W}{\Gamma(\alpha_+ + \tau n)}\} \prod_{Z_{ij}=1}^{N} p_{EMSSD}(X|\xi_j, M) \\
&\propto exp\{(\sum_{Z_{ij}=1}^{N}[I(x_i w \geq 1) + \rho_{1w})(\sum_{w=1}^{W} log(\alpha_{jw} - \tau_{jw})) \\
&\quad + (\sum_{Z_{ij}=1}^{N}[I(x_i w \geq 1)x_{iw} + \rho_{2w})(\sum_{w=1}^{W} log(\beta_j w)) \\
&\quad + k \times log\{\frac{\Gamma(\alpha_+)\tau^D}{\Gamma(\alpha_+ + \tau n)} + \sum_{Z_{ij}=1}^{N} \frac{\Gamma(\alpha_+)\tau^W}{\Gamma(\alpha_+ + \tau n_i)}\}
\end{aligned}
\tag{64}
$$

Once the sample $\mathcal{X}$ is known, the posterior hyperparameters can be fixed, we fix $\rho_{1w} = 1, \rho_{2w} = 1$, $k = 1$ and $\eta = 1$. Having the posterior in hand, we can propose the algorithm for finite mixture model of EMSSD. In Algorithm 8, $\xi_j = [\alpha_{j1}, \beta_{j1}., \tau_{j1}, \cdots, \alpha_{jW}, \beta_{jW}, \tau_{jW}]$, and we take the K-

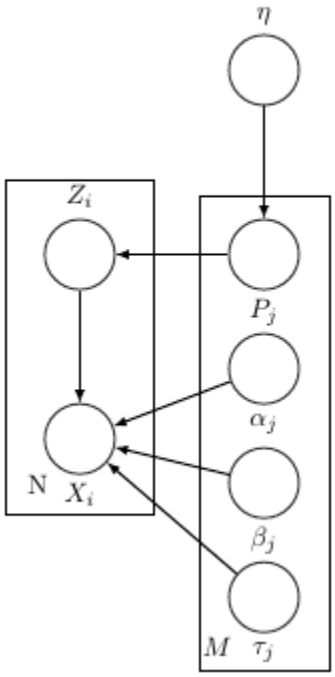

Figure 2.4: Graphical Model Representation of the Finite EMSSD.

means and the method of Moment (MOM) [57] for initializing model parameters α. We initialize β with a constant proportion vector and τ as a vector one. Choosing proposal distribution is significant part in M-H steps [23, 52]. As the model parameters satisfy $0 < \alpha_{jw} \ll 1$ and $0 < \beta_{jw} < 1$, we choose the Beta distribution and Gamma distribution as the proposal distributions for α_{jw}, β_{jw} and the Inverse Gamma distribution for τ.

$$\alpha_{jw} \sim \mathcal{B}(\alpha, \sigma_1), \tau \sim invG(\tau, \sigma_2), \beta \sim Gamma(\beta, \sigma_3) \tag{65}$$

The algorithm computation complexity for one iteration is $O(NK)$.

2.3.7 Learning Algorithm for Infinite Mixture Model of EMSSD

EMSSD is redcued to EMSD, when we set $\tau = 1$. Furthermore, EMSSD shows better experimental results in real applications. Thus, we only extend EMSSD to infinite mixture. We find that taking the prior (Eq. 63) and the posterior (Eq. 64) for EMSSD parameters in infinite mixture

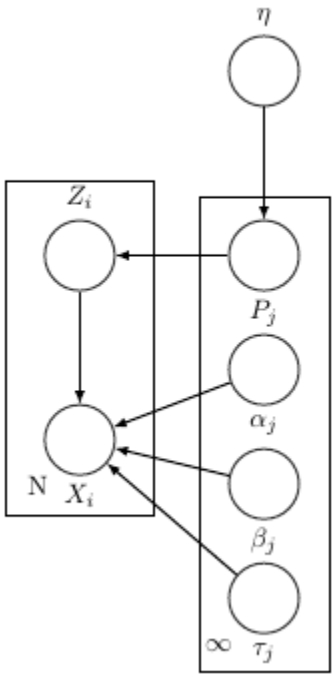

Figure 2.5: Graphical Model Representation of the Infinite EMSSD.

model, we can obtain a superior performance in real applications. Thus, we directly use them to the infinite mixture model. Then, the complete algorithm can be presented:

Algorithm 9 Infinite EMSSD (InfEMSSD)

Initialization: Using MOM to initialize model parameters
Input: A data set $\mathcal{X} = \{X_1 \cdots X_N\}$, each is W-dimensional sparse count data
output: Θ
for $t = 1 \cdots$:

(1) Generate $\mathbf{Z^t}$ from Eq.31 with Monte Carlo sampling approximation

(2) Update the number of represented components

(3) Generate parameters η from Eq. 27 with adaptive reject sampling (ARS)

(4) Generate weight parameters P^t from $Dir(\eta/M + n_1, \cdots, \eta/M + n_M)$

(5) Update α, β, τ in FinEMSSD Metropolis-Hasting(M-H) algorithm

Metropolis-Hasting(M-H) algorithm:

(1) Generate $\widetilde{\xi}_j$ from $q(\widetilde{\xi}_j|\xi_j^{t-1})$ and $u \sim U[0,1]$

(2) compute $r = \frac{\pi(\widetilde{\xi}_j|\mathbf{M},\mathbf{X})q(\xi_j^{t-1}|\widetilde{\xi}_j)}{\pi(\xi_j^{t-1}|\mathbf{M},\mathbf{X})q(\widetilde{\xi}_j|\xi_j^{t-1})}$

(3) if $r < u$ then: $\xi^t = \widetilde{\xi}$ else: $\xi^t = \xi^{t-1}$

Chapter 3

Experimental Results

In this section, we aim at comparing the proposed algorithms and their corresponding finite mixture models learned with a deterministic approach using EM algorithm in different data clustering applications. The first experiment and second one concentrate on textual data for sentiment analysis and fake news detection. The last one considers images data for distinguishing male and female faces. All experiments were conducted using optimized python code on Intel (R) Core (TM) i7-9750H processor PC with Windows 10 Enterprise Service Pack 1 operating system with a 16 GB main memory. The results that we will present in the following subsections represent the average over 20 runs of the different learning algorithms.

3.1 Text Documents Clustering

In this section, we want to test the performance of the proposed framework for text classification problems.

3.1.1 Pre-processing in Text documents

For all datasets, we did the following pre-processing before providing the experimental results of our framework:

- Text should be lowercase.
- Non-alphabetic characters should be removed.

- Stop words should be removed.
- To represent a text document, we apply a count data vector, which correlates to the frequency to which a word appears.

3.1.2 Text Sentiment Analysis

Sentiment analysis, also called opinion mining, involves analyzing evaluations, attitudes, and emotions, expressed in a piece of text, towards entities such as products, services, or movies [6]. In our experiment, we classify whether a whole opinion document expresses a positive or negative sentiment. The challenges in sentiment analysis, as a text clustering application, include that the reviews are usually limited in length, have many misspellings, and shortened forms of words. Thus, the vocabulary size is immense, and the count vector that represents each review will be highly sparse. The experiment used large data set of IMDB movies review with two labels: negative and positive, and the experiment result is based on comparing recall, precision, and F-Measure values. We collect 50,000 samples from each IMDB review of different labels, totaling 76,340 unique words. We compare the proposed algorithms with other methods, such as EGDM mixture model [62], EMBL mixture model [60], EMSSD mixture model [61] that have been proposed for modeling count data. The results are shown in Tables 3.1, according to the F-Measure in this table,

Table 3.1: IMDB Movie Reviews.

Method	Precision	Recall	F-Measure
FinEMBL-MCMC	84.72	87.88	86.27
FinEMGD-MCMC	85.16	88.73	87.03
FinEMSD-MCMC	83.85	85.66	83.64
FinEMSSD-MCMC	86.14	83.84	84.96
InfEMBL-MCMC	89.18	88.93	89.06
InfEMGD-MCMC	88.68	88.49	88.58
InfEMSSD-MCMC	**88.57**	**89.60**	**89.08**
EGDM -EM	81.36	85.55	83.59
EMBL-EM	83.75	84.60	84.17
EMSSD-EM	82.96	83.01	82.98

we can note that the proposed approaches outperform other compared models and approaches.

3.1.3 Covid-19 Fake News Detection

In 2020, a new virus swept the world and it brought many disasters to the world. In addition to the epidemic, the authenticity of news related to it from the Internet has become very important. The data set considered here contains 947 tweets that are related to Covid-19 information, and that have been already divided into two classes, one contains real news and the other contains fake news. We take all of the samples and identify the most often used 1000 unique words in 947 tweets related to Covid-19 information. From Table 3.2, our proposed algorithms still show excellent performance

Table 3.2: CON-19 Fake News Detection.

Method	Accuracy
FinEMSSD-MCMC	85.04
FinEMSD-MCMC	83.21
FinEMGD-MCMC	86.48
FinEMBL-MCMC	86.24
InfEMSSD-MCMC	86.78
InfEMGD-MCMC	**87.45**
InfEMBL-MCMC	86.26
EMGD -EM	86.50
EMBL-EM	83.75
EMSSD-EM	84.54

in the fake news detection task. Compared with other approaches and models, InfEMGD-MCMC yields the best accuracy of 87.45 %, and FinEMGD-MCMC also reaches 86.48 %. Compared with finite mixture models, the performance of our infinite mixture models show higher accuracy rate.

3.2 Images Clustering

In this section, we apply the proposed framework to test its performance in real-world image classification tasks.

3.2.1 Feature Extraction in Images

The Bag of words was originally used in text classification to represent documents as features vectors, and its basic idea is to assume that a text, ignoring its word order and grammar and syntax, is simply viewed as a collection of words and each word in the text is independent [28, 31]. Likewise,

it can be used in images as a bag of visual words. The specific process is as follows:

- We convert M images to a uniform size and then extract SIFT features for each image. Each SIFT feature is represented by a 128-dimensional vector, and we assumed that a total of N features are extracted from M images.
- Using the K-Means algorithm to divide the N objects into K clusters to have high similarity within clusters and low similarity between clusters.
- There are K cluster centers (visual words), and the distance from each SIFT feature of each image to this visual word is calculated and mapped to the visual word with the closest distance (i.e., the corresponding word frequency of that visual word + 1).

3.2.2 Human Face Gender Recognition

Figure 3.1: AR Database

In this experiment, we use two standard and challenging face recognition databases. The first database is the AR face database, which has 4000 color images corresponding to 126 people's faces (70 men and 56 women). Images feature frontal view faces with different facial expressions, illumination conditions, and occlusions (sunglasses and scarf).

Figure 3.2: Caltech Database

The second database is Caltech faces by California Institute of Technology, consists of 450 face images of around 27 unique people (both genders) with different lighting/expressions/backgrounds (sample images are shown in Fig. 3.2). We apply bag of feature (BOF) for representing the image vectors where SIFT has been used for feature extraction, treating the local image patches as the visual equivalent of individual words.

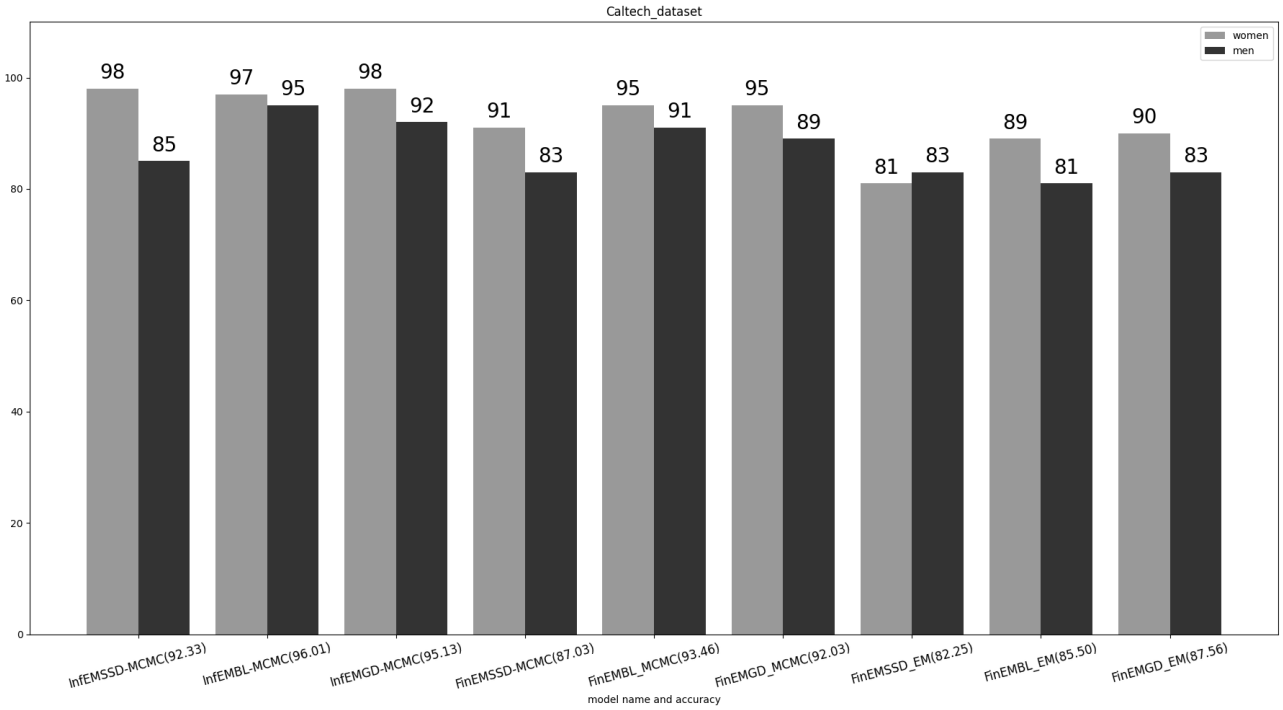

Figure 3.3: Intraclass Accuracy for Proposed Models in Caltech Database .

Fig. 3.4 and Fig 3.3 show that our proposed approaches permit good discrimination. The intraclass accuracy performance for the AR using proposed approaches is shown in Fig. 3.4. We note that InfEMSSD-MCMC shows superior performance in distinguishing women class (97%) from men class (94%) and InfEMBL-MCMC achieves 96.01% in Caltech data set as we can see in Fig. 3.3. Overall, all of our proposed models and algorithms ensure an accuracy above 85 $\%$ in this application. Compared with the EM algorithm, our proposed MCMC algorithms show higher accuracy with the corresponding models.

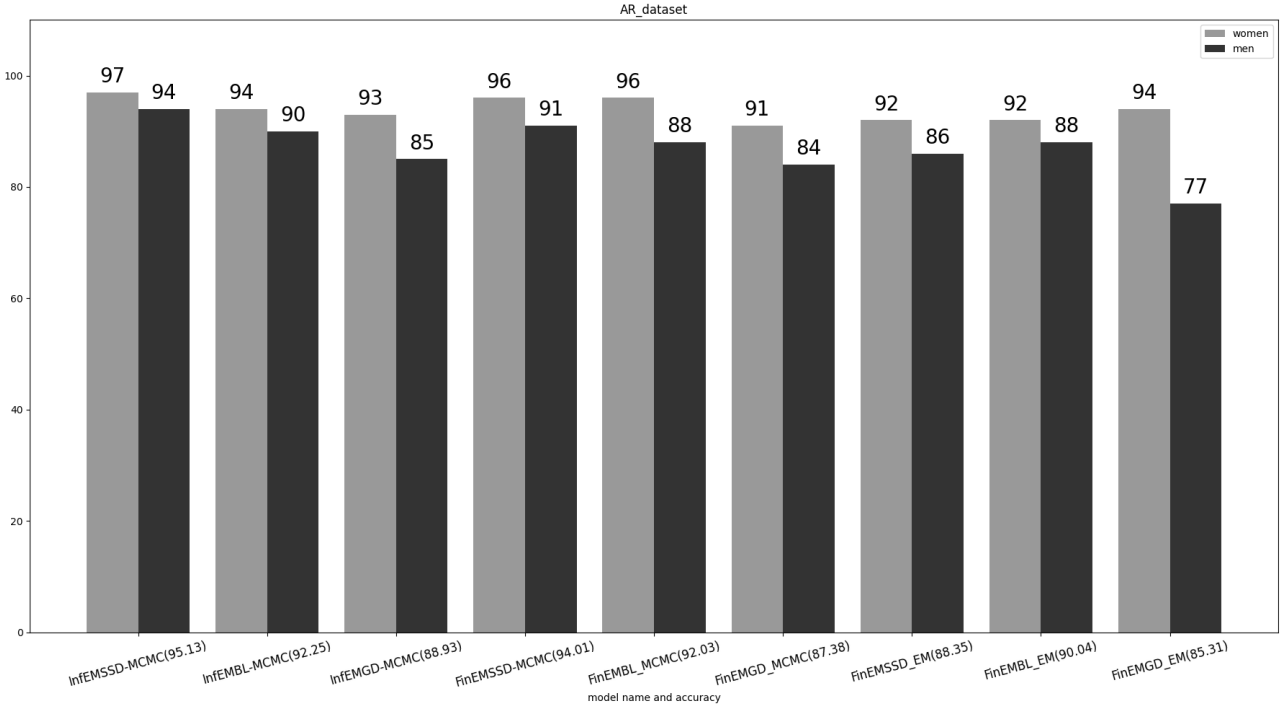

Figure 3.4: Intraclass Accuracy for Proposed Models in AR Database.

Chapter 4

Conclusion and Future Work

4.1 Conclusion

In this thesis, we have proposed learning approaches for finite mixtures of EMGD, EMBL, and EMSSD based on the development of conjugate prior-posterior distributions and the Monte Carlo simulation techniques of Gibbs sampling mixed with a M-H step. Generally, with the help of prior information and the stochastic aspect of the simulation in Gibbs sampling, Bayesian inference ensures the escape from local maximum. Moreover, we propose extensions based on infinite mixtures which model well the structure of the data. Our proposed approaches offer excellent modeling capabilities as shown by the experimental results, which involve text sentiment analysis, fake news detection and human face recognition, compared to the widely used maximum likelihood-based approach.

4.2 Future Work

Our framework still has some drawbacks as follows:

- The selection of the proposal distribution has a great impact on the final experimental results, so we had to spend a lot of time on tuning proposal distribution to achieve good experimental results. A better approach to investigate in the future is to use the particle M-H technique based on gradient and Hessian information for the posterior to construct the common proposal

distribution [22].

- The high computational complexity of the proposed inference led to slow convergence. In the future, we will concentrate on replacing classical M-H by the Scalable M-H algorithm [21]. This scheme is based on combination of factorized acceptance probabilities, procedures of Bernoulli processes, and control variate idea. It can be used to reduce the computational complexity by discovering in advance the sampling points that may be rejected.

- Apply the proposed algorithms and models to more complex tasks.

Appendix A

A.1 Proof of the Posterior in Finite Mixture of EMGD

$$\begin{aligned}\pi(\xi_j|\mathbf{M},\mathbf{X}) &\propto \pi(\xi_j)\prod_{Z_{ij}=1} EGDM(X_i|\xi_j)\\ &= exp(\sum_{w=1}^{W}\rho_l log\frac{\alpha_w\beta_w}{\alpha_w+\beta_w})\prod_{Z_{ij}=1}\{(\prod_{W:x_W\geq 1}x_w^{-1})\prod_{W:x_W\geq 1}\frac{\Gamma(z_w)}{\Gamma(x_w+z_w)}n\\ &\times\{exp\sum_{w=1}^{W}I(x_w>=1)log\frac{\alpha_w\beta_w}{\alpha_w+\beta_w}\}\}\end{aligned} \tag{66}$$

Removing the equation parts which is only related with data set $\mathcal{X}$, because it does not have an effect on the calculation in M-H step.

$$\begin{aligned}\pi(\xi_j|\mathbf{M},\mathbf{X}) &\propto \pi(\Theta_j)\prod_{Z_{ij}=1} EGDM(X_i|\Theta_j)\\ &\propto exp[\sum_{w=1}^{W}log\frac{\alpha_w\beta_w}{\alpha_w+\beta_w}(\rho_w+\sum_{Z_{ij}=1}^{N}I(x_{iw}\geq 1))]\\ &\times\prod_{Z_{ij}=1}^{N}(\prod_{w:x_{iw}\geq 1}x_{iw}^{-1}\frac{\Gamma(z_{iw})}{\Gamma(x_{iw}+z_{iw})}n!)\end{aligned} \tag{67}$$

A.2 Proof of the Posterior in Finite Mixture of EMBL

$$
\begin{aligned}
\pi(\xi_j|\mathbf{M},\mathbf{X}) &\propto \pi(\xi_j)\prod_{Z_{ij}=1} EMBL(X_i|\xi_j) \\
&= exp[\sum_{w=1}^{W}\rho_d log(\alpha_d) + k(log(\frac{\Gamma(S)\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{\Gamma(S+n)\Gamma(\alpha^{'}+\beta^{'})}))] \\
&\times \prod_{Z_{ij}=1}\{(\prod_{d:x_d>=1} x_d^{-1})n!\frac{\Gamma(S)\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{\Gamma(S+n)\Gamma(\alpha^{'}+\beta')} \times exp\{\sum_{d=1}^{D} I(x_d \le 1)log(\alpha_d)\}\} \\
&= \prod_{Z_{ij}=1}(\prod_{d:x_d>=1} x_d^{-1}n!)exp[\sum_{w=1}^{W} log(\alpha_w)(\rho_w + \sum_{Z_{ij}=1}^{N} I(x_{iw}\ge 1)) \\
&+ k(log(\frac{\Gamma(S)\Gamma(\alpha_j^{'})\Gamma(\beta_j^{'})\alpha_j}{\Gamma(S+n)\Gamma(\alpha_j^{'}+\beta_j^{'})}))
\end{aligned}
\tag{68}
$$

We remove the equations which are only related with data set $\mathcal{X}$.

For the fact that:

$$
\Gamma(S+n) = \Gamma(S)(S)\times(S+1)\cdots(S+n-1) \tag{69}
$$

So we have:

$$
\begin{aligned}
&\propto exp[\sum_{w=1}^{W} log(\alpha_w)(\rho_w + \sum_{Z_{ij}=1}^{N} I(x_{iw}\ge 1)) \\
&+ k(log(\frac{\Gamma(\alpha_j^{'})\Gamma(\beta_j^{'})\alpha_j}{(S)\times(S+1)\cdots(S+n-1)\Gamma(\alpha_j^{'}+\beta_j^{'})})) \\
&+ \sum_{i=1,z_{ij}=1}(log(\frac{\Gamma(\alpha^{'})\Gamma(\beta^{'})\alpha}{(S)\times(S+1)\cdots(S+n-1)\Gamma(\alpha^{'}+\beta^{'})}))]
\end{aligned}
\tag{70}
$$

A.3 Conditional Posterior of InfEMGD hyperparmeters

In EMGD, the conditional posteriors become:

$$
\begin{aligned}
p(\epsilon|\cdots) &\propto p(\epsilon)\prod_{j=1}^{M} p(\alpha_j|\delta,\epsilon) \\
&\lambda exp(-\lambda\epsilon) \\
&\prod_{j=1}^{M} \times \frac{\Gamma(\delta+\epsilon)}{\Gamma(\delta)\Gamma(\epsilon)}\alpha_j^{\delta-1}(1-\alpha_j)^{\epsilon-1}
\end{aligned} \tag{71}
$$

$$
\begin{aligned}
p(\delta|\cdots) &\propto p(\delta)\prod_{j=1}^{M} p(\alpha_j|\delta,\epsilon) \\
&\frac{\Gamma(\varsigma+\upsilon)}{\Gamma(\varsigma)\Gamma(\upsilon)}\delta^{\varsigma-1}(1-\varsigma)^{\upsilon-1} \\
&\times \frac{\Gamma(\delta+\epsilon)}{\Gamma(\delta)\Gamma(\epsilon)}\alpha_j^{\delta-1}(1-\alpha_j)^{\epsilon-1}
\end{aligned} \tag{72}
$$

$$
\begin{aligned}
p(\varrho|\cdots) &\propto p(\varrho)\prod_{j=1}^{M} p(\beta_j|\varpi,\varrho) \\
&= \iota exp(-\iota\varrho) \\
&\times \prod_{j=1}^{M} \frac{\Gamma(\varpi+\varrho)}{\Gamma(\varpi)\Gamma(\varrho)}\beta_j^{\varpi-1}(1-\beta_j)^{\varrho-1}
\end{aligned} \tag{73}
$$

$$
\begin{aligned}
p(\varpi|\cdots) &\propto p(\varpi))\prod_{j=1}^{M} p(\beta_j|\varpi,\varrho) \\
&= \frac{\Gamma(\kappa+\omega)}{\Gamma(\kappa)\Gamma(\omega\varpi^{\kappa-1}(1-\kappa)^{\omega-1}} \\
&\times \prod_{j=1}^{M} \frac{\Gamma(\varpi+\varrho)}{\Gamma(\varpi)\Gamma(\varrho)}\beta_j^{\varpi-1}(1-\beta_j)^{\varrho-1}
\end{aligned} \tag{74}
$$

A.4 Conditional Posterior of InfEMBL hyperparmeters

In EMBL, the form of $p(\epsilon|\cdots)$ and $p(\delta|\cdots)$ are same in Eq. 71 and Eq. 72. For the conditional posteriors of ϱ and ϖ, we have:

$$\begin{aligned} p(\varrho|\cdots) &\propto p(\varrho)\prod_{j=1}^{M} p(\beta_j|\varpi,\varrho) \\ &= \iota exp(-\iota\varrho)\prod_{j=1}^{M}\frac{\varrho^{\varpi}exp(-\varrho/\beta_j)}{\Gamma(\varpi)\beta_j^{\varpi-1}} \end{aligned} \tag{75}$$

$$\begin{aligned} p(\varpi|\cdots) &\propto p(\varpi)\prod_{j=1}^{M} p(\beta_j|\varpi,\varrho) \\ &= \frac{\omega^{\kappa}exp(-\omega/\varpi)}{\Gamma(\kappa)\varpi^{\kappa-1}}\prod_{j=1}^{M}\frac{\varrho^{\varpi}exp(-\varrho/\beta_j)}{\Gamma(\varpi)\beta_j^{\varpi-1}} \end{aligned} \tag{76}$$

Bibliography

[1] J. Aitchison. The statistical analysis of compositional data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 44(2):139–177, 1982.

[2] R. Alsuroji, N. Zamzami, and N. Bouguila. Model selection and estimation of a finite shifted-scaled dirichlet mixture model. *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 707–713, 2018.

[3] O Amayri and N. Bouguila. A bayesian analysis of spherical pattern based on finite langevin mixture. *Appl. Soft Comput.*, 38:373–383, 2016.

[4] M Amirkhan, N. Manouchehri, and N. Bouguila. Birth-death mcmc approach for multivariate beta mixture models in medical applications. *Advances and Trends in Artificial Intelligence. Artificial Intelligence Practices*, pages 285–296, 2021.

[5] T. L. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in bipolymers. *Proceedings. International Conference on Intelligent Systems for Molecular Biology*, 2:28–36, 1994.

[6] L. Batista and S. Ratté. Multi-classifier system for sentiment analysis and opinion mining. *International Conference on Advances in Social Networks Analysis and Mining*, pages 96–100, 2014.

[7] L. M. Berliner. *Bayesian Statistics: An Introduction*. Edward Arnold, London, 1997.

[8] C. M. Bishop. *Pattern recognition*. springer, 2006.

[9] N. Bouguila. Clustering of count data using generalized dirichlet multinomial distributions. *IEEE Transactions on Knowledge and Data Engineering*, 20(4):462–474, 2008.

[10] N. Bouguila. Count data modeling and classification using finite mixtures of distributions. *IEEE Transactions on Neural Networks*, 22(2):186–198, 2011.

[11] N. Bouguila and W. ElGuebaly. On discrete data clustering. In Takashi Washio, Einoshin Suzuki, Kai Ming Ting, and Akihiro Inokuchi, editors, *Advances in Knowledge Discovery and Data Mining, 12th Pacific-Asia Conference, PAKDD 2008, Osaka, Japan, May 20-23, 2008 Proceedings*, volume 5012 of *Lecture Notes in Computer Science*, pages 503–510. Springer, 2008.

[12] N. Bouguila and D. Ziou. A dirichlet process mixture of generalized dirichlet distributions for proportional data modeling. *IEEE Transactions on Neural Networks*, 21(1):107–122, 2010.

[13] N. Bouguila, D. Ziou, and R. I. Hammoud. On bayesian analysis of a finite generalized dirichlet mixture via a metropolis-within-gibbs sampling. *Pattern Analysis & Applications*, 12(2):151–166, 2009.

[14] N. Bouguila, D. Ziou, and H. Riad I. A bayesian non-gaussian mixture analysis: Application to eye modeling. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[15] S. P. Brooks. On bayesian analyses and finite mixtures for proportions. *Statistics and Computing*, 11(2):179–190, 2001.

[16] L. D. Brown. Fundamentals of statistical exponential families with applications in statistical decision theory. *Lecture Notes-Monograph Series*, 9:100–279, 1986.

[17] K. L. Caballero, B. Joel, and A. Ram. The generalized dirichlet distribution in enhanced topic detection. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, page 773–782, New York, NY, USA, 2012. Association for Computing Machinery.

[18] L. Cai. High-dimensional exploratory item factor analysis by a metropolis–hastings robbins–monro algorithm. *Psychometrika*, 75(1):33–57, 2010.

[19] P. Cerchiello and G. Paolo. Dirichlet compound multinomials for text modelling. *Applied Mathematics*, pages 2089–2097, 01 2012.

[20] K. W. Church and W. A. Gale. Poisson mixtures. *Natural Language Engineering*, 1(2):163–190, 1995.

[21] R. Cornish, P. Vanetti., A. Bourchard, G. Deligiannidis, and A. Doucet. Scalable Metropolis-Hastings for exact Bayesian inference with large datasets. *Proceedings of the 36th International Conference on Machine Learning*, 97:1351–1360, 2019.

[22] J. Dahlin, F. Lindsten., and T. Schön. Particle metropolis–hastings using gradient and hessian information. *Statistics and Computing*, 25:81–92, 2015.

[23] S. Daniel and D. Gianola. *Likelihood, Bayesian and MCMC methods in quantitative genetics*. Springer, New York, 2002.

[24] A. Dasgupta. *The Exponential Family and Statistical Applications*, pages 583–612. Springer New York, 2011.

[25] I. S. Dhillon and D. S. Modha. Concept decompositions for large sparse text data using clustering. *Machine Learning*, 42(1/2):143–175, 2001.

[26] J. G. Dias and M. Wedel. An empirical comparison of em, sem and mcmc performance for problematic gaussian mixture likelihoods. *Statistics and Computing*, 14(4):323–332, 2004.

[27] T. Elguebaly and N. Bouguila. Bayesian learning of generalized gaussian mixture models on biomedical images. In Friedhelm Schwenker and Neamat El Gayar, editors, *Artificial Neural Networks in Pattern Recognition, 4th IAPR TC3 Workshop, ANNPR 2010, Cairo, Egypt, April 11-13, 2010. Proceedings*, volume 5998 of *Lecture Notes in Computer Science*, pages 207–218. Springer, 2010.

[28] T. Elguebaly and N. Bouguila. *A Bayesian Method for Infrared Face Recognition*, pages 123–138. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[29] T. Elguebaly and N. Bouguila. A bayesian approach for the classification of mammographic masses. In *2013 Sixth International Conference on Developments in eSystems Engineering*, pages 99–104, 2013.

[30] C. Elkan. Clustering documents with an exponential-family approximation of the dirichlet compound multinomial distribution. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 289–296. Association for Computing Machinery, 2006.

[31] W. Fan and N. Bouguila. Online facial expression recognition based on finite beta-liouville mixture models. In *2013 International Conference on Computer and Robot Vision*, pages 37–44, 2013.

[32] C. Forbes, E. Merran, H. Nicholas, and P. Brian. *Statistical distributions.* John Wiley and Sons, 2011.

[33] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Journal of Applied Statistics*, 20(5-6):25–62, 1993.

[34] R. K. Hankin. A generalization of the dirichlet distribution. *Journal of Statistical Software*, 33(11):1–18, 2010.

[35] W. Keith Hastings. Monte carlo sampling methods using markov chains and their applications. *Oxford University Press*, 1970.

[36] J. Herve, D. Matthijs, and S. Cordelia. On the burstiness of visual elements. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1169–1176, 2009.

[37] M. A. Islam and R. I. Chowdhury. *Exponential Family of Distributions*, pages 23–30. Springer Singapore, 2017.

[38] A. K. Jain. Data clustering: 50 years beyond k-means. *Pattern Recognition Letters*, 31(8):651–666, 2010.

[39] S. M. Katz. Distribution of content words and phrases in text and language modelling. *Natural Language Engineering*, 2(1):15–59, 1996.

[40] S. Kotz, N. Balakrishnan, and N. L. Johnson. *Continuous Multivariate Distributions*. Wiley Online Library, New York, 2014.

[41] E. Levitan and G. T. Herman. A maximum a posteriori probability expectation maximization algorithm for image reconstruction in emission tomography. *IEEE Transactions on Medical Imaging*, 6(3):185–192, 1987.

[42] E. Madsen, D. Kauchak, and C. Elkan. Modeling word burstiness using the dirichlet distribution. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05, page 545–552. Association for Computing Machinery, 2005.

[43] D. Margaritis and S. Thrun. A bayesian multiresolution independence test for continuous variables. *the Seventeenth Conference on Uncertainty in Artificial Intelligence (UAI2001)*, pages 346–353, 2013.

[44] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *American Institute of Physics*, 21(6):1087–1092, 1953.

[45] F. Najar, N. Zamzami, and N. Bouguila. Fake news detection using bayesian inference. *2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI)*, pages 389–394, 2019.

[46] F. Najar, N. Zamzami, and N. Bouguila. Recognition of human interactions in feature films based on infinite mixture of edcm. *2020 International Symposium on Networks, Computers and Communications (ISNCC)*, pages 1–6, 2020.

[47] C. E. Rasmussen. The infinite gaussian mixture model. *Advances in Neural Information Processing Systems*, 12:554–560, 1999.

[48] C. Robert. *The Bayesian choice: from decision-theoretic foundations to computational implementation*, volume 2. Springer, 2007.

[49] Z. Song, S. Ali., and N. Bouguila. Bayesian learning of infinite asymmetric gaussian mixture models for background subtraction. In *Image Analysis and Recognition - 16th International*

Conference, ICIAR 2019, Waterloo, ON, Canada, August 27-29, 2019, Proceedings, Part I, volume 11662 of *Lecture Notes in Computer Science*, pages 264–274. Springer, 2019.

[50] X. Su, N. Bouguila, and N. Zamzami. Covid-19 news clustering using mcmc-based learing of finite emsd mixture models. *The International FLAIRS Conference Proceedings*, 34, 2021.

[51] N. Tomasev and M. Radovanovic. *Clustering Evaluation in High-Dimensional Data*, pages 71–107. Springer International Publishing, Cham, 2016.

[52] K. E. Train. *Discrete Choice Methods with Simulation*, pages 76–96. Cambridge University Press, 2 edition, 2009.

[53] E. G. Tsionas. Bayesian inference for multivariate gamma distributions. *Statistics and Computing*, 14(3):223–233, 2004.

[54] Tzu-Tsung. W. Generalized dirichlet distribution in bayesian analysis. *Applied Mathematics and Computation*, 97(2-3):165–181, 1998.

[55] Tzu-Tsung. W. A bayesian approach employing generalized dirichlet priors in predicting microchip yields. *Journal of the Chinese Institute of Industrial Engineers*, 22(3):210–217, 2005.

[56] Tzu-Tsung. W. Alternative prior assumptions for improving the performance of naïve bayesian classifiers. *Data Mining and Knowledge Discovery*, 18(2):183–213, 2009.

[57] Tzu-Tsung. W. Parameter estimation for generalized dirichlet distributions from the sample estimates of the first and the second moments of random variables. *Computational Statistics & Data Analysis*, 54(7):1756–1765, 2010.

[58] A. Wong and H. John. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society*, 28(1):100–108, 1979.

[59] N. Zamzami and N. Bouguila. A novel scaled dirichlet-based statistical framework for count data modeling: Unsupervised learning and exponential approximation. *Pattern Recognition*, 95:36–47, 2019.

[60] N. Zamzami and N. Bouguila. High-dimensional count data clustering based on an exponential approximation to the multinomial beta-liouville distribution. *Information Sciences*, 524:116–135, 2020.

[61] N. Zamzami and N. Bouguila. Probabilistic modeling for frequency vectors using a flexible shifted-scaled dirichlet distribution prior. *ACM Trans. Knowl. Discov. Data*, 14(6):35–69, 2020.

[62] N. Zamzami and N. Bouguila. Sparse count data clustering using an exponential approximation to generalized dirichlet multinomial distributions. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–14, 2020.