The Thirty-Third International
FLAIRS Conference (FLAIRS-33)

Improving the EDCM Mixture Model with Expectation Propagation

Xavier Sumba,[1] Nuha Zamzami,[2,3] Nizar Bouguila[2]
[1]Department of Electrical and Computer Engineering, Concordia University, Montreal, QC., Canada
[2]Concordia Institute for Information Systems Engineering, Concordia University, Montreal, QC., Canada
[3]Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
xavier.sumba93@ucuenca.edu.ec, n_zamz@encs.concordia.ca, nizar.bouguila@concordia.ca

Abstract

Bayesian inference is crucial to challenging scenarios that involve complex probabilistic models, which are usually intractable. In this work, we develop an expectation propagation approach to learn finite mixture models of EDCMs. The EDCM (Elkan 2006) is an exponential-family approximation to the widely used Dirichlet Compound Multinomial distribution and has been shown to offer excellent modeling capabilities in the case of sparse count data. Expectation propagation is a deterministic approach that provides accurate approximations to the full posterior and allows to include prior beliefs in the model as opposed to the maximum-likelihood method, which provides point estimates only. We evaluate the efficiency of our framework on several datasets for sentiment analysis and shape recognition. Our proposed model shows comparable to superior results to other approaches in the literature.

1 Introduction

Document clustering is widely considered in a variety of applications, such as text retrieval. The words in text documents usually exhibit appearance dependencies, *i.e.*, if a word w appears once, it is more probable that the same word w will appear again. This phenomenon is denominated as burstiness, which has shown to be addressed using Dirichlet Compound Multinomial (DCM) distribution (Madsen, Kauchak, and Elkan 2005). Similarly, different distributions had been used in order to model burstiness while preserving conjugacy, but the estimation of their parameters of these models is considerably slow, especially in high-dimensional spaces. Thus, taking into account the sparsity and high-dimensionality of text data, (Elkan 2006) proposed the EDCM, which approximates the DCM as a member of the exponential family. Indeed, EDCM has shown to offer fast parameter learning and a helpful intuition for the study of the burstiness phenomenon.

Parameter learning is one of the encountered challenges in mixture models, and typically the maximum-likelihood method via the expectation-maximization (EM) algorithm has been used for learning the parameters of an EDCM mixture model (Elkan 2006). Despite that the maximum-likelihood method has shown fast parameter learning, this approach suffers from numerous inconveniences, such as providing a point estimate, which impacts on the accuracy of the learned model. Moreover, the appropriate number of components has to be known in advance, which can be approached by using information-theory based techniques such as Minimum Message Length criterion (MML) (Zamzami and Bouguila 2019). Deterministic Bayesian inference techniques (*e.g.* variational inference or expectation propagation) allow good approximation of the full posterior. Recently, (Najar, Zamzami, and Bouguila 2019) proposed the use of a sampling method, *i.e.*, Markov Chain Monte Carlo (MCMC), for learning an EDCM mixture and have shown the importance of having priors, outperforming previous results. However, sampling methods are computationally expensive. This work studies the application of the Bayesian framework for learning the EDCM mixture model. In particular, we propose an approach for learning a finite EDCM mixture model parameters using Expectation Propagation (EP) (Minka 2001). EP, a deterministic approximate inference framework, has shown to be more accurate than methods such as variational inference and MCMC.

The rest of this paper is organized as follows. First, Section 2 revisits the Exponential-family Approximation to DCM Distribution (EDCM) model upon which our work is built on. Next, in Section 3, we outline the EDCM mixture model, describe the expectation propagation approach, and derive the complete learning framework. Section 4 describes our experimental setup and evaluation of our proposed method. Finally, we conclude the paper in Section 5.

2 The Exponential-family Approximation to DCM Distribution

We are given a dataset $\mathcal{X}$ with D samples $\mathcal{X} = \{\mathbf{x_i}\}_{i=1}^{D}$, each $\mathbf{x}_i$ is a vector of count data (*e.g.* a document or an image, represented as a vector of word frequencies or visual words respectively). In (Madsen, Kauchak, and Elkan 2005), the authors proposed a generative model to deal with burstiness phnomenon by introducing a prior Dirichlet distribution with parameters $\boldsymbol{\alpha}$ to the Multinomial model. They define a new marginal distribution by integrating out $\boldsymbol{\theta}$, obtaining a discrete distribution known as the Dirichlet Compound

Multinomial (DCM) or multivariate Polya distribution.

$$\mathcal{DCM}(\mathbf{x} \mid \boldsymbol{\alpha}) = \frac{n!}{\prod_{w=1}^{V} x_w!} \frac{\Gamma(s)}{\Gamma(s+n)} \prod_{w=1}^{V} \frac{\Gamma(x_w+\alpha_w)}{\Gamma(\alpha_w)} \tag{1}$$

where $s = \sum_{w=1}^{V} \alpha_w$ is the sum of the Dirichlet distribution parameters.

Given that text documents representation is very sparse because not every word appears in most of the documents, in (Elkan 2006), the authors noted that using only the non-zero values of $\mathbf{x}$ is computationally efficient. Moreover, the parameter α_w of the DCM distribution is small for most words, $\alpha_w \ll 1$. Thus, replacing $\frac{\Gamma(x_w+\alpha_w)}{\Gamma(\alpha_w)}$ by $\Gamma(x_w)\alpha_w$ and using the fact that $\Gamma(x_w) = (x_w - 1)!$ leads to an approximation of the DCM distribution known as EDCM. We replace α with β in order to follow the same notation as in (Elkan 2006):

$$\mathcal{EDCM}(\mathbf{x} \mid \beta) = n! \frac{\Gamma(s)}{\Gamma(s+n)} \prod_{w:x_w \geq 1} \frac{\beta_w}{x_w} \tag{2}$$

3 The proposed framework

3.1 Expectation Propagation Approach

Expectation Propagation (EP) (Minka 2001) EP handles partitioned data and combines partitions iteratively through message passing. Having the latent variable $\boldsymbol{\Theta}$, EP approximates a target distribution $p(\boldsymbol{\Theta} \mid \mathcal{X})$, which is commonly the posterior, with a global approximation $q(\boldsymbol{\Theta})$ that belongs to the exponential family. The choice of q depends on the problem but it has to be a simple approximating distribution that can be fitted using small refinements. To apply EP, first split the posterior in D sites $p(\boldsymbol{\Theta} \mid \mathcal{X}) \propto p_0(\boldsymbol{\Theta}) \prod_i^D p_i(\mathbf{x}_i \mid \boldsymbol{\Theta})$; the initial site t_0 is commonly represented with the prior distribution and the remaining p_i sites represent the contribution of each term to the likelihood. The approximating distribution must admit a similar factorization, *i.e.* $q(\boldsymbol{\Theta}) \propto \prod_i^D \tilde{p}_i(\boldsymbol{\Theta})$. Therefore, the goal of EP is to refine each of the approximating sites such that they capture the contribution of each of the likelihood sites to the posterior, *i.e.* $\tilde{p}_i(\boldsymbol{\Theta}) \approx p_i(\mathbf{x} \mid \boldsymbol{\Theta})$. Each approximating site has to be initialized and belong to the exponential family. Consequently, each site is refined to create a cavity distribution, $q^{\setminus i}(\boldsymbol{\Theta}) \propto q(\boldsymbol{\Theta})/\tilde{p}_i(\boldsymbol{\Theta})$, by dividing the global approximation over the current approximate site.

Additionally, in order to approximate each site, we introduce a new tilted distribution which consists in the product of the cavity distribution and the current site $q_i^*(\boldsymbol{\Theta}) \propto p_i(\boldsymbol{\Theta}) q^{\setminus i}(\boldsymbol{\Theta})$. Subsequently, a new posterior is found by minimizing the Kullback Leibler divergence $D_{KL}(q_i^*(\boldsymbol{\Theta}) \,||\, q^{new}(\boldsymbol{\Theta}))$ such that $\tilde{p}_i(\boldsymbol{\Theta}) \approx p_i(\mathbf{x} \mid \boldsymbol{\Theta})$. This minimization is equivalent to match the moments of those distributions We can also notice that this updating scheme creates a coupling for the approximating factors, so updates must be iterated. Finally, the revised approximate site is updated by removing the remaining terms from the current approximation $\tilde{p}_i(\boldsymbol{\Theta}) \propto q^{new}(\boldsymbol{\Theta})/q^{\setminus i}(\boldsymbol{\Theta})$.

3.2 Mixture-based Clustering Model

Here, we state the settings for a finite EDCM mixture model and develop a mathematical framework for learning the mixture using expectation propagation. We assume that we are given D documents drawn from an $\mathcal{EDCM}$ distribution, and each $\mathbf{x}_i$ document is composed of V words. $K \geq 1$ represents the number of mixture components or clusters. Thus, a document is drawn from its respective component j as follows: $\mathbf{x_i} \sim \mathcal{EDCM}(\boldsymbol{\beta}_j)$.

Consequently, a latent variable $\mathcal{Z} = \{\mathbf{z}_i\}_{i=1}^D$ is introduced for each $\mathbf{x}_i$ document in order to represent the component assignment. We posit a Multinomial distribution for the component assignment such that $\mathbf{z}_i \sim Mult(1, \boldsymbol{\pi})$ where $\boldsymbol{\pi} = \{\pi_j\}_{j=1}^K$ represents the mixing weights, and they are subject to the constraints $0 < \pi_j < 1$ and $\sum_j \pi_j = 1$. In other words, $\mathbf{z}_i$ is a K-dimensional indicator vector containing a value of one when document $\mathbf{x}_i$ belongs to the component j, and zero otherwise. Note that in this setting the value of $z_{ij} = 1$ acts as the selector of the component that generates $\mathbf{x}_i$ document with parameter $\boldsymbol{\beta}_j$; hence, $p(\mathbf{z}_i \mid \boldsymbol{\pi}) = \pi_j$. The full posterior can be written as $p(\boldsymbol{\pi}, \boldsymbol{\beta} \mid \mathcal{X}) \propto p(\boldsymbol{\pi}) p(\boldsymbol{\beta}) \prod_i^D \sum_j^K \pi_j p(\mathbf{x}_i \mid \boldsymbol{\beta}_j)$

3.3 Parameters Learning

In this section, we describe the learning approach using EP algorithm. We partition the likelihood in D sites and start by defining an ith approximating site for each of the latent variables ($\boldsymbol{\pi}$ and $\boldsymbol{\beta}$). First, we assign a Dirichlet distribution with parameter $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$ as a prior for the mixing weights since it fits properly the constraints imposed by the model and works as a nice prior for the mixing weights $\boldsymbol{\pi}$ that holds conjugacy properties.

$$\tilde{p}_i(\boldsymbol{\pi} \mid \boldsymbol{\alpha}_i) \propto \prod_{j=1}^{K} \pi_j^{\alpha_{ij}-1} \tag{3}$$

For the $\boldsymbol{\beta}$ variable of the EDCM mixture, we adopt a Gaussian distribution, which leads to an intractable distribution since $\tilde{p}(\boldsymbol{\pi})$ is a Dirichlet distribution. However, this setting has been used successfully to approximate a Beta and Dirichlet distribution (Ma and Leijon 2010; Fan and Bouguila 2014). Additionally, a Gaussian distribution not only allows analytically tractable calculations but also capture correlation for the values of $\boldsymbol{\beta}_j$. Hence, we select for the approximating site of $\boldsymbol{\beta}_j$ a Gaussian distribution with mean $\mathbf{m}_{ij}$ and precision matrix $\boldsymbol{\Lambda}_{ij}^{-1}$ for each j component.

$$\tilde{p}_i(\boldsymbol{\beta}) \propto \prod_{j=1}^{K} \exp\left(-\frac{1}{2}(\boldsymbol{\beta}_j - \mathbf{m}_{ij})^{\intercal} \boldsymbol{\Lambda}_{ij} (\boldsymbol{\beta}_j - \mathbf{m}_{ij})\right) \tag{4}$$

The EDCM mixture model posterior $p(\boldsymbol{\pi}, \boldsymbol{\beta})$ can be factorized in D sites, one for each document i with priors $p(\boldsymbol{\pi})$ and $p(\boldsymbol{\beta})$. Additionally, after defining the approximate sites, we compute the approximate posterior $q(\boldsymbol{\pi}, \boldsymbol{\beta} \mid \boldsymbol{\alpha}', \mathbf{m}', \boldsymbol{\Lambda}'^{-1})$ by getting the product of D approximate sites, where $\boldsymbol{\alpha}'$, $\mathbf{m}'$, and $\boldsymbol{\Lambda}'$ are the parameters of the posterior distribution and can be calculated using Eq. 5.

$$\alpha'_j = \sum_i^D \alpha_{i,j} - D$$
$$\boldsymbol{\Lambda}'_j = \sum_i^D \boldsymbol{\Lambda}_{i,j}; \quad \mathbf{m}'_j = \boldsymbol{\Lambda}_j^{'-1} \left(\sum_i^D \boldsymbol{\Lambda}_{i,j} \mathbf{m}_{i,j} \right) \tag{5}$$

In order to create a refinement for the approximate site $p_i(\boldsymbol{\pi}, \boldsymbol{\beta})$, we introduce a cavity distribution $q^{\setminus i}(\boldsymbol{\pi}, \boldsymbol{\beta})$ by deleting the contribution of the ith site. Thus, the cavity distribution has parameters $\boldsymbol{\alpha}^{\setminus i}$, $\boldsymbol{\Lambda}^{\setminus i}$, and $\mathbf{m}^{\setminus i}$ as shown in Eq. 6, and it is calculated as follows: $q(\boldsymbol{\pi}, \boldsymbol{\beta})/\tilde{p}_i(\boldsymbol{\pi}, \boldsymbol{\beta})$.

$$\alpha_j^{\setminus i} = \alpha'_j - \alpha_{i,j} + 1; \quad \boldsymbol{\Lambda}_j^{\setminus i} = \boldsymbol{\Lambda}'_j - \boldsymbol{\Lambda}_{i,j}$$
$$\mathbf{m}_j^{\setminus i} = \boldsymbol{\Lambda}_j^{\setminus i-1} \left(\boldsymbol{\Lambda}_j^{'} \mathbf{m}_j^{'} - \boldsymbol{\Lambda}_{i,j} \mathbf{m}_{i,j} \right) \tag{6}$$

Then, we incorporate the contribution of the ith site to the cavity distribution, resulting in a tilted distribution $q^*(\boldsymbol{\pi}, \boldsymbol{\beta})$ that is an updated posterior. We normalize this new posterior using a normalizing factor Z_i to guarantee that it is a proper distribution. The normalizing factor can be then calculated by integrating out $\boldsymbol{\pi}$ and $\boldsymbol{\beta}$. However, the integration of the normalization factor is not possible since the integral is intractable. Thus, we propose to solve this integral via Monte Carlo sampling, where we take S samples from $\boldsymbol{\beta}_s \sim \mathcal{N}(\mathbf{m}^{\setminus i}, \boldsymbol{\Lambda}^{\setminus i-1})$. Therefore, after rewriting the normalization factor, the following expression is obtained:

$$Z_i(\boldsymbol{\alpha}^{\setminus i}, \mathbf{m}_j^{\setminus i}, \boldsymbol{\Lambda}_j^{\setminus i}) = \textstyle\sum_j^K \frac{\alpha_j^{\setminus i}}{\sum_j^K \alpha_j^{\setminus i}} \mathbb{E}_{p(\boldsymbol{\beta}_j)} \left[\mathcal{EDCM}(\mathbf{x}_i \mid \boldsymbol{\beta}_j) \right] \tag{7}$$

Finally, we minimize the KL divergence between the tilted distribution and the approximate posterior $D_{KL}(q_i^*(\boldsymbol{\pi}, \boldsymbol{\beta}) \quad || \quad q^{new}(\boldsymbol{\pi}, \boldsymbol{\beta}))$. This minimization is achieved by calculating the partial derivative of $\log Z_i$ with respect to the parameters of the model and matching its respective moments. After matching the sufficient statistics of $\mathbb{E}_{q^*(\boldsymbol{\pi},\boldsymbol{\beta})} \left[\nabla_{\alpha_j^{\setminus i}} \log Dir(\boldsymbol{\pi}) \right]$, $\mathbb{E}_{q^*(\boldsymbol{\pi},\boldsymbol{\beta})} \left[\nabla_{m_j^{\setminus i}} \log \mathcal{N}(\boldsymbol{\beta}_j) \right]$, and $\mathbb{E}_{q^*(\boldsymbol{\pi},\boldsymbol{\beta})} \left[\nabla_{\boldsymbol{\Lambda}_j^{\setminus i}} \log \mathcal{N}(\boldsymbol{\beta}_j) \right]$, we update the parameters of the approximate posterior $q^{new}(\boldsymbol{\pi}, \boldsymbol{\beta})$ using Eqs. (8-10).

$$\Psi(\alpha_j^{'}) - \Psi(\textstyle\sum_j^K \alpha_j^{'}) = \nabla_{\alpha_j^{\setminus i}} \log Z_i - \Psi(\sum_j^K \alpha_j^{\setminus i}) + \Psi(\alpha_j^{\setminus i}) \tag{8}$$

$$\mathbf{m}'_j = \boldsymbol{\Lambda}_j^{\setminus i-1} (\nabla_{\mathbf{m}_j^{\setminus i}} \log Z_i + \boldsymbol{\Lambda}_j^{\setminus i} \mathbf{m}_j^{\setminus i}) \tag{9}$$

$$\boldsymbol{\Lambda}_j^{'} = -2\nabla_{\boldsymbol{\Lambda}_j^{\setminus i}} \log Z_i + \boldsymbol{\Lambda}_j^{\setminus i} - \mathbf{m}_j^{'} \mathbf{m}_j^{'\intercal} + 2\mathbf{m}_j^{'} \mathbf{m}_j^{\setminus i\intercal} - \mathbf{m}_j^{\setminus i} \mathbf{m}_j^{\setminus i\intercal} \tag{10}$$

The gradient of $\log Z_i$, can be calculated analytically using Eq. (7). Finally, we reuse the updated approximate posterior and remove the cavity distribution in order to obtain the update for the current approximate site $\tilde{p}_i$, where the ith site have the following parameters:

$$\alpha_{i,j} = \alpha'_j - \alpha_j^{\setminus i} + 1; \quad \boldsymbol{\Lambda}_{i,j} = \boldsymbol{\Lambda}'_j - \boldsymbol{\Lambda}_j^{\setminus i} \tag{11}$$

$$\mathbf{m}_{i,j} = \left(\boldsymbol{\Lambda}_j^{'-1} - \boldsymbol{\Lambda}_j^{\setminus i-1} \right) \left(\boldsymbol{\Lambda}_j^{'} \mathbf{m}_j^{'} - \boldsymbol{\Lambda}_j^{\setminus i} \mathbf{m}_j^{\setminus i} \right) \tag{12}$$

This procedure is repeated for all the D documents and iterated until a certain level of convergence is reached. The values of the mixing weights can be approximated by calculating its expectation with respect to the approximating posterior $\mathbb{E}_q \left[\pi_j \right] = \alpha_j^{'} / \sum_{j=1}^K \alpha_j^{'}$.

3.4 A Note on Initialization and Learning Algorithm

We initialize each approximate site such that $\tilde{p}_i(\boldsymbol{\pi}, \boldsymbol{\beta}) \rightarrow 1$. The approximate posterior is initialized with the values of the prior $q(\boldsymbol{\pi}, \boldsymbol{\beta}) = \tilde{p}_0(\boldsymbol{\pi}, \boldsymbol{\beta})$. For instance, we initialize the mixing weights uniformly, thus we consider a symmetric Dirichlet prior $\tilde{p}_0(\boldsymbol{\pi})$ with parameter value $1/K$. Consequently, for the prior $p(\boldsymbol{\beta})$, we follow an adaptation of the method of moments (MoM) described in (Bouguila and Ziou 2007). We compute an initial $\boldsymbol{\beta}_j$ and calculate its statistics as follows: 1) apply K-means clustering; 2) apply MoM for the EDCM distribution to each j component found; 3) calculate $\mathbf{m}_{0,j}$ and $\boldsymbol{\Lambda}_{0,j}$. It is possible to encode any prior information in the mixing weights (*i.e.* the means of the k-means clusters). Nevertheless, for the EDCM parameter $\boldsymbol{\beta}$, we find that the MoM restricts the values of $\boldsymbol{\beta}$ to be small and positive while sampling from a Gaussian distribution. This initialization scheme helps the proposed framework to stabilize while fitting the values of $\beta_{j,w} \ll 1$.

4 Results

4.1 Sentiment Analysis

We evaluate the proposed framework in a Sentiment Analysis task, using three benchmark datasets: 1) Amazon Review Polarity; 2) Yelp review Polarity; 3) IMDB Movie Reviews.

Experimental setup For each j component, at inference time, we set all values to zero except the diagonal ones from the precision matrix $\boldsymbol{\Lambda}_{*j}^{-1}$ for computational simplicity. Additionally, we take $S = 100$ samples from $\mathcal{N}(\mathbf{m}^{\setminus i}, \boldsymbol{\Lambda}^{\setminus i-1})$ and force all values to be positive. For every dataset, we analyze the effect of pre-processing. In other words, we examine whether pre-processing helps the mixture to fit the data better. We performed the following pre-processing for all datasets: 1) lowercase all text; 2) remove non-alphabetical characters; 3) remove stop words; 4) lemmatize text.

Results We apply the proposed framework to all the datasets described in the above section. We compare our approach with an EDCM mixture model using maximum-likelihood (ML) for learning its parameters as reported in (Elkan 2006). Additionally, we evaluate the effect of pre-processing text documents when using the proposed method since in latent models (such as LDA), it has been shown that common pre-processing steps have no impact on the obtained results.Thus, we evaluate our parameter learning method where pre-processing is involved (EP-P) and raw text (EP-NP). We evaluate our results in terms of precision and recall as shown in Table 1.

Amazon Review Polarity dataset our framework completely outperforms the maximum-likelihood estimation by

Table 1: Results on the three text datasets. Comparison using precision and recall for every inference method. ML: maximum-likelihood; EP-P: expectation propagation + pre-processing; EP-NP: expectation propagation + raw text.

		Dataset		
Metrics		Amazon	Yelp	IMDB
Precision	ML	80.65	**89.25**	78.54
	EP-P	84.84	74.26	78.60
	EP-NP	**86.91**	80.50	**86.36**
Recall	ML	80.88	89.28	**89.33**
	EP-P	81.23	**93.83**	78.45
	EP-NP	**84.82**	78.60	85.94

$\sim$6% and $\sim$4% improvement for precision and recall respectively, and thus, achieving 86.91% and 84.82%. Additionally, we notice that pre-processing causes a bad effect on the model instead of helping infer the right cluster assignments. For *Yelp Review Polarity* dataset our approach outperforms the maximum-likelihood approach in terms of recall, meaning that the EP model is more confident at assigning the right clusters. Finally, for the *IMDB movie review* EP surpasses ML in terms of precision by a large margin $\sim$9%.

4.2 Shape recognition

For Shape recognition, we use the Swedish leaf dataset (Söderkvist 2001) that contains 15 different types of leaves. We evaluate with 26 and 39 clusters (i.e. $K = 26, K = 39$). Mixture components π_j with very small values are ignored.

Experimental setup The leaf dataset contains 585 images, each corresponding to a specific Swedish specie. Each image size is 128×128. For each image, we extracted 200 discrete features. In order to extract features from the leaves images, we use shape context in which an object is assumed to be essentially captured by a finite set of its N points sampled from the internal or external contours on the object. A shape context is a descriptor for each point, which captures the distribution of the remaining points relative to the current one. We sample 200 points from internal and external image boundaries and create a vector of visual words.

Results We compare the mixture of EDCM model with both ML and EP inference methods and report performance in terms of accuracy (see Table 2) using the leaf dataset. The proposed model improves the accuracy of the leaf dataset. The EDCM mixture with ML gets an accuracy of 94.45 while results with EP improves accuracy by 3.67%, obtaining 98.12 when using 26 components. On the other hand, we obtain a lower accuracy with a greater number of components $K = 39$. Consequently, with a number of clusters smaller than 26 we get an average accuracy of ~ 78.

5 Conclusions

In this paper, we propose the use of Expectation Propagation (EP) to learn a finite EDCM mixture model instead of the maximum-likelihood (ML), and as a result, incorporating some advantages that the Bayesian framework provides. EP is used to learn the model parameters, and additionally, we notice that the number of clusters can be determined by ignoring or merging components with very small values of the expected mixing weights. Moreover, we propose a simple but optimal initialization scheme in order to meet the restrictions that the approximation of the DCM distribution is subject to. Given that we use the Bayesian framework, some other sources of prior information can be encoded in the model. Finally, we demonstrate the efficacy of our framework by evaluating it in sentiment analysis and shape recognition tasks. Results show the validity of our framework and obtaining comparable and superior results as opposed to using ML estimation in terms of clustering performance.

Table 2: Results for shape recognition in the leaf dataset. Comparison using accuracy for every inference method. ML: maximum-likelihood; EP: expectation propagation.

Inference	Accuracy
ML	94.45
EP ($K = 26$)	**98.12**
EP ($K = 39$)	88.76

References

Bouguila, N., and Ziou, D. 2007. Unsupervised learning of a finite discrete mixture: Applications to texture modeling and image databases summarization. *Journal of Visual Communication and Image Representation* 18(4):295–309.

Elkan, C. 2006. Clustering documents with an exponential-family approximation of the dirichlet compound multinomial distribution. In *Proceedings of the 23rd international conference on Machine learning*, 289–296. ACM.

Fan, W., and Bouguila, N. 2014. Non-gaussian data clustering via expectation propagation learning of finite dirichlet mixture models and applications. *Neural processing letters* 39(2):115–135.

Ma, Z., and Leijon, A. 2010. Expectation propagation for estimating the parameters of the beta distribution. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2082–2085. IEEE.

Madsen, R. E.; Kauchak, D.; and Elkan, C. 2005. Modeling word burstiness using the dirichlet distribution. In *Proceedings of the 22nd international conference on Machine learning*, 545–552. ACM.

Minka, T. P. 2001. Expectation propagation for approximate bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, 362–369.

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

Söderkvist, O. 2001. Computer vision classification of leaves from swedish trees.

Zamzami, N., and Bouguila, N. 2019. Model selection and application to high-dimensional count data clustering: via finite EDCM mixture models. *Applied Intelligence* 49(4):1467–1488.