Soft Comput (2016) 20:979–990
DOI 10.1007/s00500-014-1557-5

METHODOLOGIES AND APPLICATION

Variational learning of hierarchical infinite generalized Dirichlet mixture models and applications

Wentao Fan · Hassen Sallay · Nizar Bouguila · Sami Bourouis

Published online: 16 December 2014

Abstract Data clustering is a fundamental unsupervised learning task in several domains such as data mining, computer vision, information retrieval, and pattern recognition. In this paper, we propose and analyze a new clustering approach based on both hierarchical Dirichlet processes and the generalized Dirichlet distribution, which leads to an interesting statistical framework for data analysis and modelling. Our approach can be viewed as a hierarchical extension of the infinite generalized Dirichlet mixture model previously proposed in Bouguila and Ziou (IEEE Trans Neural Netw 21(1):107–122, 2010). The proposed clustering approach tackles the problem of modelling grouped data where observations are organized into groups that we allow to remain statistically linked by sharing mixture components. The resulting clustering model is learned using a principled variational Bayes inference-based algorithm that we have developed. Extensive experiments and simulations, based on two challenging applications namely images categorization and web service intrusion detection, demonstrate our model usefulness and merits.

Keywords Mixture models · Hierarchical Dirichlet processes · Generalized Dirichlet distribution · Variational Bayes · Image categorization · Web services

Communicated by V. Loia.

W. Fan
Department of Computer Science and Technology, Huaqiao University, Xiamen, China
e-mail: fwt@hqu.edu.cn

H. Sallay
College of Computer and Information Systems, Umm Al-Qura University, Makkah, Saudi Arabia
e-mail: hmsallay@imamu.edu.sa

N. Bouguila (✉)
Concordia Institute for Information Systems Engineering (CIISE), Concordia University, Montreal, QC, Canada
e-mail: nizar.bouguila@concordia.ca

S. Bourouis
Taif University, Taif, Saudi Arabia
e-mail: s.bourouis@tu.edu.sa

1 Introduction

A large literature on the analysis, classification, and clustering of data is now available (Jain et al. 2004; Law et al. 2005; Banerjee et al. 2004; Xiang et al. 2008). This is fueled by the fact that a large number of real-world problems can be easily approached as categorization tasks. The number of approaches and techniques is overwhelming. In particular, many generative models have been proposed for probabilistic classification (Kahn 2004). The Gaussian mixture model (Li et al. 2006; McLachlan and Peel 2000; Lu and Yao 2005) is perhaps the most well-known and widely used generative approach. Although the Gaussian mixture has proven to be an effective approach for data clustering when the partitions are Gaussian, it is known that this approach can fail in several real-world problems when the data are clearly non-Gaussian as deeply discussed and reported in the literature (Bouguila and Ziou 2006, 2007; Shoham et al. 2003). Indeed, a major difficulty of generative models is that they require the choice of a parent distribution to model the data. Thus, much of the work on finite mixture models has focused on determining appropriate choices for different kinds of data (e.g. discrete, binary, continuous, directional, etc.). For instance, we have shown recently that the Dirichlet (Bouguila and Ziou 2005) and generalized Dirichlet (GD) (Bouguila and Ziou 2006, 2007) distributions could be good alternatives to the Gaussian for data modelling. The main advantage of the

GD over the Dirichlet is that it has a more general covariance structure which makes it more flexible and useful for a variety of real life applications from different disciplines. Therefore, in this work, in lieu of considering the classic Gaussian assumption, we base our work on the GD mixture.

The statistical models proposed in (Bouguila and Ziou 2006, 2007) have been based on finite GD mixtures and then needed to address challenging learning problems including model selection (i.e. determination of the optimal number of components), the choice of an initialization approach to avoid convergence to saddle points and over- or under-fitting problems. A recently introduced model provides a new strategy where the number of components is supposed to be infinite which allows to overcome problems related to model selection (Bouguila and Ziou 2010). This approach, based on Dirichlet processes, appears currently to offer the best results in terms of efficiency and model selection accuracy. Moreover, it avoids classic problems related to initialization sensitivity and convergence.

This work is an effort to deal with the classification of data in a flexible fashion and can be viewed as an major extension of the framework previously developed in (Bouguila and Ziou 2010). Indeed, in this paper, we go a step further by focusing on the problem of modelling grouped data where observations are organized into groups that we allow to remain statistically linked by sharing mixture components. Our solution is based on the adoption of a hierarchical nonparametric Bayesian framework namely hierarchical Dirichlet process (Teh et al. 2006; Teh and Jordan 2010), which is motivated by its promising results shown recently when modelling grouped data generated from various fields. Appropriate learning is a key issue when using generative models and is a quite difficult task, especially when dealing with high-dimensional data. Recently, variational Bayes learning has been shown to be an efficient alternative to both frequentist and purely Bayesian techniques (Fan and Bouguila 2013; Fan et al. 2013). We, therefore, develop a principled variational approach to learn the parameters of our hierarchical Dirichlet process of GD distributions model, which can be viewed as the second major contribution of this research work. The third contribution concerns the challenging problems that we have tackled to illustrate the usefulness of our approach namely images categorization and web service intrusion detection.

The rest of the article is organized as follows. In Sect. 2 we present the basic hierarchical Dirichlet process mixture model and its adoption to GD distributions. The model's learning approach, based on variational inference, as well as the complete fitting algorithm are detailed in Sect. 3. Section 4 presents and discusses the experiments and simulations conducted to assess the viability and efficiency of the proposed model. Section 5 concludes the paper.

2 Hierarchical Dirichlet process mixture of GD distributions

In this section, we introduce our hierarchical Dirichlet process mixture model of GD distributions, which may also be referred to as the hierarchical infinite GD mixture model.

2.1 Hierarchical Dirichlet process mixture model

The hierarchical Dirichlet process is built on the Dirichlet process (Korwar and Hollander 1973; Ferguson 1983) with a Bayesian hierarchy in which the base distribution of the Dirichlet process is itself distributed according to a Dirichlet process. A two-level hierarchical Dirichlet process model can be defined as the following: given a grouped data set $\mathcal{X}$ with M groups, where each group is associated with a Dirichlet process G_j, and this indexed set of Dirichlet processes $\{G_j\}$ shares a global (or base) distribution G_0 which is itself distributed according to a Dirichlet process with the base distribution H and concentration parameter γ:

$$\begin{aligned} G_0 &\sim \mathrm{DP}(\gamma, H) \\ G_j &\sim \mathrm{DP}(\lambda, G_0) \qquad \text{for each } j,\ j \in \{1, \ldots, M\} \end{aligned} \tag{1}$$

where j is an index for each group of the data set. Please notice that the above hierarchical Dirichlet process can be readily extended to contain more than two levels.

In this work, the hierarchical Dirichlet process is represented using the stick-breaking construction (Sethuraman 1994; Ishwaran and James 2001). In the global-level construction, the global measure G_0 is distributed according to the Dirichlet process $\mathrm{DP}(\gamma, H)$ as

$$\begin{aligned} G_0 &= \sum_{k=1}^{\infty} \psi_k \delta_{\Omega_k} \qquad \psi_k = \psi'_k \prod_{s=1}^{k-1} (1 - \psi'_s) \\ \psi'_k &\sim \mathrm{Beta}(1, \gamma) \qquad \Omega_k \sim H \end{aligned} \tag{2}$$

where δ_{Ω_k} is an atom at Ω_k, and $\{\Omega_k\}$ is a set of independent random variables drawn from H. The stick-breaking variable ψ_k satisfies $\sum_{k=1}^{\infty} \psi_k = 1$, and is obtained by recursively breaking a unit length stick into an infinite number of pieces such that the size of each successive piece is proportional to the rest of the stick. Since G_0 is the base distribution of the Dirichlet processes G_j and has the stick-breaking representation as shown in Eq. (2), the atoms Ω_k are shared among all $\{G_j\}$ and only differ in weights according to the property of Dirichlet process (Teh et al. 2006).

Motivated by (Wang et al. 2011), we also apply the conventional stick-breaking representation to construct each group-level Dirichlet process G_j as

$$G_j = \sum_{t=1}^{\infty} \pi_{jt} \delta_{\varpi_{jt}} \qquad \pi_{jt} = \pi'_{jt} \prod_{s=1}^{t-1} (1 - \pi'_{js})$$
$$\pi'_{jt} \sim \text{Beta}(1, \lambda) \qquad \varpi_{jt} \sim G_0 \tag{3}$$

where $\delta_{\varpi_{jt}}$ are group-level atoms at ϖ_{jt}, $\{\pi_{jt}\}$ is a set of stick-breaking weights which satisfies $\sum_{t=1}^{\infty} \pi_{jt} = 1$. Since ϖ_{jt} is distributed according to the base distribution G_0, it takes on the value Ω_k with probability ψ_k. Next, we introduce a binary latent variable W_{jtk} as an indicator variable, such that $W_{jtk} \in \{0, 1\}$, $W_{jtk} = 1$ if ϖ_{jt} maps to the global-level atom Ω_k which is indexed by k; otherwise, $W_{jtk} = 0$. Thus, we can have $\varpi_{jt} = \Omega_k^{W_{jtk}}$. Accordingly, group-level atoms ϖ_{jt} do not need to be explicitly represented. The indicator variable $\mathbf{W} = (W_{jt1}, W_{jt2}, \ldots)$ is distributed as

$$p(\mathbf{W}|\boldsymbol{\psi}) = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \prod_{k=1}^{\infty} \psi_k^{W_{jtk}} \tag{4}$$

Since $\boldsymbol{\psi}$ is a function of $\boldsymbol{\psi}'$ according to the stick-breaking construction of the Dirichlet process as shown in Eq. (2), we can rewrite $p(\mathbf{W})$ as

$$p(\mathbf{W}|\boldsymbol{\psi}') = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \prod_{k=1}^{\infty} \left[\psi'_k \prod_{s=1}^{k-1} (1 - \psi'_s) \right]^{W_{jtk}} \tag{5}$$

The prior distribution of $\boldsymbol{\psi}'$ is a specific Beta distribution according to Eq. (2)

$$p(\boldsymbol{\psi}') = \prod_{k=1}^{\infty} \text{Beta}(1, \gamma_k) = \prod_{k=1}^{\infty} \gamma_k (1 - \psi'_k)^{\gamma_k - 1} \tag{6}$$

For the grouped data set $\mathcal{X}$, let i index the observations within each group j. We assume that each variable θ_{ji} is a factor corresponding to an observation X_{ji}, and the factors $\boldsymbol{\theta}_j = (\theta_{j1}, \theta_{j2}, \ldots)$ are distributed according to the Dirichlet process G_j, one for each j. Then, the likelihood function can be defined as

$$\theta_{ji}|G_j \sim G_j$$
$$X_{ji}|\theta_{ji} \sim F(\theta_{ji}) \tag{7}$$

where $F(\theta_{ji})$ represents the distribution of the observation X_{ji} given θ_{ji}. The base distribution H of G_0 provides the prior distribution for the factors θ_{ji}. This setting is known as the hierarchical Dirichlet process mixture model, in which each group is associated with a mixture model, and the mixture components are shared among these mixture models due to the sharing of atoms Ω_k among all $\{G_j\}$.

Since each factor θ_{ji} is distributed according to G_j based on Eq. (7), it takes the value ϖ_{jt} with probability π_{jt}. We then place a binary indicator variable $Z_{jit} \in \{0, 1\}$ for θ_{ji}. That is, $Z_{jit} = 1$ if θ_{ji} is associated with component t and maps to the group-level atom ϖ_{jt}; otherwise, $Z_{jit} = 0$. Thus, we have $\theta_{ji} = \varpi_{jt}^{Z_{jit}}$. Since ϖ_{jt} also maps to the global-level atom Ω_k as we mentioned previously, we then have $\theta_{ji} = \varpi_{jt}^{Z_{jit}} = \Omega_k^{W_{jtk} Z_{jit}}$. The indicator variable $\mathbf{Z} = (Z_{ji1}, Z_{ji2}, \ldots)$ is distributed as

$$p(\mathbf{Z}|\boldsymbol{\pi}) = \prod_{j=1}^{M} \prod_{i=1}^{N} \prod_{t=1}^{\infty} \pi_{jt}^{Z_{jit}} \tag{8}$$

Since $\boldsymbol{\pi}$ is a function of $\boldsymbol{\pi}'$ according to the stick-breaking construction of the Dirichlet process as shown in Eq. (3), we then have

$$p(\mathbf{Z}|\boldsymbol{\pi}') = \prod_{j=1}^{M} \prod_{i=1}^{N} \prod_{t=1}^{\infty} \left[\pi'_{jt} \prod_{s=1}^{t-1} (1 - \pi'_{js}) \right]^{Z_{jit}} \tag{9}$$

The prior distribution of $\boldsymbol{\pi}'$ is a specific Beta distribution as described in Eq. (3) as

$$p(\boldsymbol{\pi}') = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \text{Beta}(1, \lambda_{jt}) = \prod_{j=1}^{M} \prod_{t=1}^{\infty} \lambda_{jt} (1 - \pi'_{jt})^{\lambda_{jt} - 1} \tag{10}$$

2.2 Hierarchical infinite GD mixture model

In our work, we focus on a specific form of hierarchical Dirichlet process mixture model where each observation within a group is drawn from a mixture of GD distributions. Since Dirichlet process mixture models are often referred to as infinite mixture models, the proposed model can then be considered as a hierarchical infinite GD mixture model. The motivation of choosing GD mixture is due to its good modelling performance as shown, for instance, in Bouguila and Ziou (2006, 2007), Fan and Bouguila (2013), Fan et al. (2013).

If a D-dimensional random vector $\mathbf{Y} = (Y_1, \ldots, Y_D)$ is distributed according to a GD distribution with positive parameters $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_D)$ and $\boldsymbol{\beta}_j = (\beta_1, \ldots, \beta_D)$, then its probability density function (pdf) is defined by

$$\text{GD}(\mathbf{Y}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \prod_{l=1}^{D} \frac{\Gamma(\alpha_l + \beta_l)}{\Gamma(\alpha_l)\Gamma(\beta_l)} Y_l^{\alpha_l - 1} \left(1 - \sum_{f=1}^{l} Y_f\right)^{\gamma_l} \tag{11}$$

where $\Gamma(\cdot)$ is the gamma function, $\sum_{l=1}^{D} Y_l < 1$ and $Y_l > 0$ for $l = 1, \ldots, D$, $\alpha_l > 0$, $\beta_l > 0$, $\gamma_l = \beta_l - \alpha_{l+1} - \beta_{l+1}$ for $l = 1, \ldots, D-1$, and $\gamma_D = \beta_D - 1$. It is noteworthy that GD distribution has a more general covariance structure (can be

positive or negative) than Dirichlet distribution which makes it more practical and useful.

By applying an interesting mathematical property of the GD distribution which is thoroughly discussed in Boutemedjet et al. (2009), we can use a geometric transformation to transform the original data point $\mathbf{Y}$ into another D-dimensional data point $\mathbf{X}$ with independent features as

$$\mathrm{GD}(\mathbf{X}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \prod_{l=1}^{D} \mathrm{Beta}(X_l|\alpha_l, \beta_l) \tag{12}$$

where $\mathbf{X} = (X_1, \ldots, X_D)$, $X_1 = Y_1$ and $X_l = Y_l/(1 - \sum_{f=1}^{l-1} Y_f)$ for $l > 1$, and $\mathrm{Beta}(X_l|\alpha_l, \beta_l)$ is a Beta distribution defined with parameters (α_l, β_l). By doing this, the estimation of a D-dimensional GD distribution is transformed to D estimations of one-dimensional Beta distributions which may facilitate the inference process for multidimensional data (Boutemedjet et al. 2009).

Now for the grouped data set $\mathcal{X}$ with M groups, we assume that each D-dimensional data vector $\mathbf{X}_{ji} = (X_{ji1}, \ldots, X_{jiD})$ is drawn from a hierarchical infinite GD mixture model, then we have the likelihood function of the proposed hierarchical infinite GD mixture model with latent variables as

$$p(\mathcal{X}) = \prod_{j=1}^{M} \prod_{i=1}^{N} \prod_{t=1}^{\infty} \prod_{k=1}^{\infty} \left[\prod_{l=1}^{D} \mathrm{Beta}(X_{jil}|\alpha_{kl}, \beta_{kl}) \right]^{Z_{jit} W_{jtk}} \tag{13}$$

Next, we need to place prior distributions over parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ of the Beta distribution. Although Beta distribution belongs to the exponential family and has a formal conjugate prior, it is analytically intractable. Thus, we adopt Gamma distribution to approximate the conjugate prior by assuming that these Beta parameters are statistically independent as

$$p(\boldsymbol{\alpha}) = \mathcal{G}(\boldsymbol{\alpha}|\mathbf{u}, \mathbf{v}) = \prod_{k=1}^{\infty} \prod_{l=1}^{D} \frac{v_{kl}^{u_{kl}}}{\Gamma(u_{kl})} \alpha_{kl}^{u_{kl}-1} \mathrm{e}^{-v_{kl}\alpha_{kl}} \tag{14}$$

$$p(\boldsymbol{\beta}) = \mathcal{G}(\boldsymbol{\beta}|\mathbf{g}, \mathbf{h}) = \prod_{k=1}^{\infty} \prod_{l=1}^{D} \frac{h_{kl}^{g_{kl}}}{\Gamma(g_{kl})} \beta^{g_{kl}-1} \mathrm{e}^{-h_{kl}\beta_{kl}} \tag{15}$$

where $\mathcal{G}(\cdot)$ is the Gamma distribution with positive parameters.

3 Variational model learning

Variational inference (Attias 1999; Bishop 2006) is a deterministic approximation technique that is used to find tractable approximations for posterior distributions of a variety of statistical models. The literature of the past decade is abundant with papers that have considered successfully variational learning. In this section, we propose a truncated variational inference framework to learn the stick-breaking representation of the hierarchical infinite GD mixture model. To simplify notation, we define $\Theta = (\mathbf{Z}, \mathbf{W}, \boldsymbol{\psi}', \boldsymbol{\pi}', \boldsymbol{\alpha}, \boldsymbol{\beta})$ as the set of latent variables and unknown random variables. Similarly, the set of all observed variables is represented by $\mathcal{X}$. The central idea in variational learning is to find an approximation $q(\Theta)$ for the true posterior distribution $p(\Theta|\mathcal{X})$ by maximizing the lower bound of the logarithm of the marginal likelihood $p(\mathcal{X})$. By applying Jensen's inequality, this lower bound can be found as

$$\mathcal{L}(q) = \int q(\Theta) \ln[p(\mathcal{X}, \Theta)/q(\Theta)] \mathrm{d}\Theta \tag{16}$$

In our work, we apply the truncation technique as described in (Blei and Jordan 2005) to truncate the variational approximations of global- and group-level Dirichlet processes at K and T, such that

$$\psi'_K = 1, \quad \sum_{k=1}^{K} \psi_k = 1, \quad \psi_k = 0 \ \text{ when } \ k > K \tag{17}$$

$$\pi'_{jT} = 1, \quad \sum_{t=1}^{T} \pi_{jt} = 1, \quad \pi_{jt} = 0 \ \text{ when } \ t > T \tag{18}$$

where the truncation levels K and T are variational parameters which can be freely initialized and will be optimized automatically during the learning process. Moreover, we adopt a factorial approximation to factorize $q(\Theta)$ into disjoint tractable distributions. Using the truncated stick-breaking representations and the factorization assumption, we then have

$$q(\Theta) = q(\mathbf{Z})q(\mathbf{W})q(\boldsymbol{\pi}')q(\boldsymbol{\psi}')q(\boldsymbol{\alpha})q(\boldsymbol{\beta}) \tag{19}$$

To maximize the lower bound $\mathcal{L}(q)$, we need to make a variational optimization of it with respect to each of the factors $q_i(\Theta_i)$ in turn. The general expression for the optimal solution to a specific variational factor $q_s(\Theta_s)$ is given by

$$q_s(\Theta_s) = \frac{\exp\langle \ln p(\mathcal{X}, \Theta) \rangle_{i \neq s}}{\int \exp\langle \ln p(\mathcal{X}, \Theta) \rangle_{i \neq s} \mathrm{d}\Theta} \tag{20}$$

where $\langle \cdot \rangle_{i \neq s}$ is the expectation with respect to all the distributions of $q_i(\Theta_i)$ except for $i = s$.

Therefore, the optimal solutions for the factors of the variational posterior can then be obtained by applying Eq. (20) to each of the factor, such that

$$q(\mathbf{Z}) = \prod_{j=1}^{M}\prod_{i=1}^{N}\prod_{t=1}^{T}\rho_{jit}^{Z_{jit}} \tag{21}$$

$$q(\mathbf{W}) = \prod_{j=1}^{M}\prod_{t=1}^{T}\prod_{k=1}^{K}\vartheta_{jtk}^{W_{jtk}} \tag{22}$$

$$q(\boldsymbol{\pi}') = \prod_{j=1}^{M}\prod_{t=1}^{T}\mathrm{Beta}(\pi'_{jt}|a_{jt}, b_{jt}) \tag{23}$$

$$q(\boldsymbol{\psi}') = \prod_{k=1}^{K}\mathrm{Beta}(\psi'_{k}|c_k, d_k) \tag{24}$$

$$q(\boldsymbol{\alpha}) = \prod_{k=1}^{K}\prod_{l=1}^{D}\mathcal{G}(\alpha_{kl}|\tilde{u}_{kl}, \tilde{v}_{kl}) \tag{25}$$

$$q(\boldsymbol{\beta}) = \prod_{k=1}^{K}\prod_{l=1}^{D}\mathcal{G}(\beta_{kl}|\tilde{g}_{kl}, \tilde{h}_{kl}) \tag{26}$$

where the corresponding hyperparameters in the above equations can be calculated as the following:

The hyperparameter ρ_{jit} of the factor $q(\mathbf{Z})$ is calculated by

$$\rho_{jit} = \frac{\exp(\widetilde{\rho}_{jit})}{\sum_{f=1}^{T}\exp(\widetilde{\rho}_{jif})} \tag{27}$$

where

$$\widetilde{\rho}_{jit} = \sum_{k=1}^{K}\langle W_{jtk}\rangle\sum_{l=1}^{D}\Big[(\bar{\alpha}_{kl}-1)\ln X_{jil} + (\bar{\beta}_{kl}-1)\ln(1-X_{jil}) + \widetilde{\mathcal{R}}_{kl}\Big] + \langle\ln\pi'_{jt}\rangle + \sum_{s=1}^{t-1}\langle\ln(1-\pi'_{js})\rangle \tag{28}$$

$$\begin{aligned}\widetilde{\mathcal{R}} = {} & \ln\frac{\Gamma(\bar{\alpha}+\bar{\beta})}{\Gamma(\bar{\alpha})\Gamma(\bar{\beta})} + \bar{\alpha}[\Psi(\bar{\alpha}+\bar{\beta}) - \Psi(\bar{\alpha})]\big(\langle\ln\alpha\rangle - \ln\bar{\alpha}\big) \\ & + \bar{\beta}[\Psi(\bar{\alpha}+\bar{\beta}) - \Psi(\bar{\beta})]\big(\langle\ln\beta\rangle - \ln\bar{\beta}\big) \\ & + \frac{1}{2}\bar{\alpha}^2[\Psi'(\bar{\alpha}+\bar{\beta}) - \Psi'(\bar{\alpha})]\big\langle(\ln\alpha - \ln\bar{\alpha})^2\big\rangle \\ & + \frac{1}{2}\bar{\beta}^2[\Psi'(\bar{\alpha}+\bar{\beta}) - \Psi'(\bar{\beta})]\big\langle(\ln\beta - \ln\bar{\beta})^2\big\rangle \\ & + \bar{\alpha}\bar{\beta}\Psi'(\bar{\alpha}+\bar{\beta})(\langle\ln\alpha\rangle - \ln\bar{\alpha})(\langle\ln\beta\rangle - \ln\bar{\beta})\end{aligned} \tag{29}$$

where $\Psi(\cdot)$ is the digamma function.

The hyperparameter ϑ_{jtk} of the factor $q(\mathbf{W})$ is calculated by

$$\vartheta_{jtk} = \frac{\exp(\widetilde{\vartheta}_{jtk})}{\sum_{f=1}^{K}\exp(\widetilde{\vartheta}_{jtf})} \tag{30}$$

where

$$\widetilde{\vartheta}_{jtk} = \sum_{i=1}^{N}\langle Z_{jit}\rangle\sum_{l=1}^{D}\Big[(\bar{\alpha}_{kl}-1)\ln X_{jil} + (\bar{\beta}_{kl}-1)\ln(1-X_{jil}) + \widetilde{\mathcal{R}}_{kl}\Big] + \langle\ln\psi'_k\rangle + \sum_{s=1}^{k-1}\langle\ln(1-\psi'_s)\rangle \tag{31}$$

The hyperparameters a_{jt} and b_{jt} of the factor $q(\boldsymbol{\pi}')$ are calculated by

$$a_{jt} = 1 + \sum_{i=1}^{N}\langle Z_{jit}\rangle, \qquad b_{jt} = \lambda_{jt} + \sum_{i=1}^{N}\sum_{s=t+1}^{T}\langle Z_{jis}\rangle \tag{32}$$

The hyperparameters c_k and d_k of the factor $q(\boldsymbol{\psi}')$ are calculated by

$$c_k = 1 + \sum_{j=1}^{M}\sum_{t=1}^{T}\langle W_{jtk}\rangle, \quad d_k = \gamma_k + \sum_{j=1}^{M}\sum_{t=1}^{T}\sum_{s=k+1}^{K}\langle W_{jts}\rangle \tag{33}$$

The hyperparameters $\tilde{u}_{kl}$ and $\tilde{v}_{kl}$ of the factor $q(\boldsymbol{\alpha})$ are calculated by

$$\tilde{u}_{kl} = u_{kl} + \sum_{j=1}^{M}\sum_{t=1}^{T}\langle W_{jtk}\rangle\sum_{i=1}^{N}\langle Z_{jit}\rangle\bar{\alpha}_{kl}\big[\Psi(\bar{\alpha}_{kl}+\bar{\beta}_{kl}) - \Psi(\bar{\alpha}_{kl}) + \bar{\beta}_{kl}\Psi'(\bar{\alpha}_{kl}+\bar{\beta}_{kl})(\langle\ln\beta_{kl}\rangle - \ln\bar{\beta}_{kl})\big] \tag{34}$$

and

$$\tilde{v}_{kl} = v_{kl} - \sum_{j=1}^{M}\sum_{t=1}^{T}\langle W_{jtk}\rangle\sum_{i=1}^{N}\langle Z_{jit}\rangle\ln X_{jil} \tag{35}$$

The hyperparameters $\tilde{g}_{kl}$ and $\tilde{h}_{kl}$ of the factor $q(\boldsymbol{\beta})$ are calculated by

$$\tilde{g}_{kl} = g_{kl} + \sum_{j=1}^{M}\sum_{t=1}^{T}\langle W_{jtk}\rangle\sum_{i=1}^{N}\langle Z_{jit}\rangle\bar{\beta}_{kl}\big[\Psi(\bar{\alpha}_{kl}+\bar{\beta}_{kl}) - \Psi(\bar{\beta}_{kl}) + \bar{\alpha}_{kl}\Psi'(\bar{\alpha}_{kl}+\bar{\beta}_{kl})(\langle\ln\alpha_{kl}\rangle - \ln\bar{\alpha}_{kl})\big] \tag{36}$$

and

$$\tilde{h}_{kl} = h_{kl} - \sum_{j=1}^{M}\sum_{t=1}^{T}\langle W_{jtk}\rangle\sum_{i=1}^{N}\langle Z_{jit}\rangle\ln(1-X_{jil}) \tag{37}$$

The expected values in the above formulas are given by

$$\bar{\alpha}_{kl} = \frac{\tilde{u}_{kl}}{\tilde{v}_{kl}}, \quad \bar{\beta}_{kl} = \frac{\tilde{g}_{kl}}{\tilde{h}_{kl}},$$

$$\langle Z_{jit} \rangle = \rho_{jit}, \quad \langle W_{jtk} \rangle = \vartheta_{jtk} \tag{38}$$

$$\langle \ln \alpha_{kl} \rangle = \Psi(\tilde{u}_{kl}) - \ln \tilde{v}_{kl},$$

$$\langle \ln \beta_{kl} \rangle = \Psi(\tilde{g}_{kl}) - \ln \tilde{h}_{kl} \tag{39}$$

$$\langle \ln \pi'_{jt} \rangle = \Psi(a_{jt}) - \Psi(a_{jt} + b_{jt}) \tag{40}$$

$$\langle \ln(1 - \pi'_{jt}) \rangle = \Psi(b_{jt}) - \Psi(a_{jt} + b_{jt}) \tag{41}$$

$$\langle \ln \psi'_{k} \rangle = \Psi(c_k) - \Psi(c_k + d_k) \tag{42}$$

$$\langle \ln(1 - \psi'_{k}) \rangle = \Psi(d_k) - \Psi(c_k + d_k) \tag{43}$$

Since the solutions to the variational factors are coupled together through the expected values of other factors, this optimization process can be solved in a way analogous to the EM algorithm and the complete algorithm is summarized in Algorithm 1. The convergence of this variational learning algorithm is guaranteed and can be monitored through inspection of the variational lower bound (Bishop 2006).

Algorithm 1 Variational learning of hierarchical infinite GD mixture model.

1: Choose the initial truncation levels K and T.
2: Initialize the values for hyperparameters $\lambda_{jt}, \gamma_k, u_{kl}, v_{kl}, g_{kl}, h_{kl}$.
3: Initialize the value of ρ_{jit} by K-Means algorithm.
4: **repeat**
5: *The variational E-step*:
6: Estimate the expected values in Eqs. (38–43), use the current distributions over the model parameters.
7: *The variational M-step*:
8: Update the variational solutions for each factor using Eqs. (21–26) and the current values of the moments.
9: **until** Convergence criterion is reached.

4 Experiments

4.1 Design of experiments

We evaluate the effectiveness of the proposed hierarchical infinite GD mixture model (referred to as *HInGD*) on two challenging real-life problems. The first one concerns visual scenes classification and more specifically the discrimination between different breeds of cats and dogs from images. This task is extremely challenging since cats and dogs are highly deformable and different breeds may differ only by a few subtle phenotypic details (Parkhi et al. 2012). The second application is web service intrusion detection which has attracted a lot of attention recently. We conducted our experiments using a computer with Intel's Core i7 processor @2.00 GHz. In our experiments, we initialize the global truncation level K to 600, and the group truncation level T to 100. This is because in general, the number of clusters in the global level is much larger than the one in the group level. The initial values of involved hyperparameters are set as the following: $(u_{kl}, v_{kl}, g_{kl}, h_{kl}, \lambda_{jt}, \gamma_k) = (0.25, 0.01, 0.25, 0.01, 0.1, 0.1)$. These specific choices were chosen according to our experimental results and were found convenient in our case. As a formal approach to choose the initial hyperparameters values does not exist, it may be helpful in practise to run the optimization several times using different initializations to find a good maximum since multiple maxima may exist in the variational bound. It is noteworthy that the Bayesian nature of the proposed learning algorithm makes it less sensitive to initialization as compared to frequentist techniques for instance.

4.2 Images categorization

4.2.1 Methodology

Image categorization has been the topic of extensive research in the past which may be motivated by its various applications such as object detection, recognition, and retrieval (Lamdan et al. 1988; Agarwal and Roth 2002; Matas et al. 2002; Lazebnik et al. 2004; Rasiwasia and Vasconcelos 2008). Here, we focus on a specific image categorization problem that has received some attention recently namely the classification of images representing cats and dogs (Parkhi et al. 2012). We perform this classification using the proposed *HInGD*. Our categorization methodology is summarized as following: first, we extract and normalize PCA-SIFT descriptors[1] (36-dimensional) (Ke and Sukthankar 2004) from raw images using the Difference-of-Gaussian (DoG) detector (Mikolajczyk and Schmid 2004). Then, these extracted image features are modelled using the proposed *HInGD*. Specifically, each image $\mathcal{I}_j$ is considered as a "group" and is therefore associated with an infinite mixture model G_j. Thus, each extracted PCA-SIFT feature vector X_{ji} of the image $\mathcal{I}_j$ is supposed to be drawn from the infinite mixture model G_j, where the mixture components of G_j can be considered as "visual words". A global vocabulary is constructed and is shared among all groups (images) through the common global infinite mixture model G_0 of our hierarchical model. This setting matches the desired design of a hierarchical Dirichlet process mixture model. It is noteworthy that an important step in image categorization approaches with bag-of-visual words representation is the construction of visual vocabulary. Nevertheless, most of the previously invented approaches have to apply a separate vector quantization method (such as K-means) to build the visual vocabulary, where the size of the vocabulary is normally chosen manually. In our approach, the construction of the visual vocabulary is part of our hierarchical Dirichlet process mix-

[1] PCA-SIFT: http://www.cs.cmu.edu/~yke/pcasift.

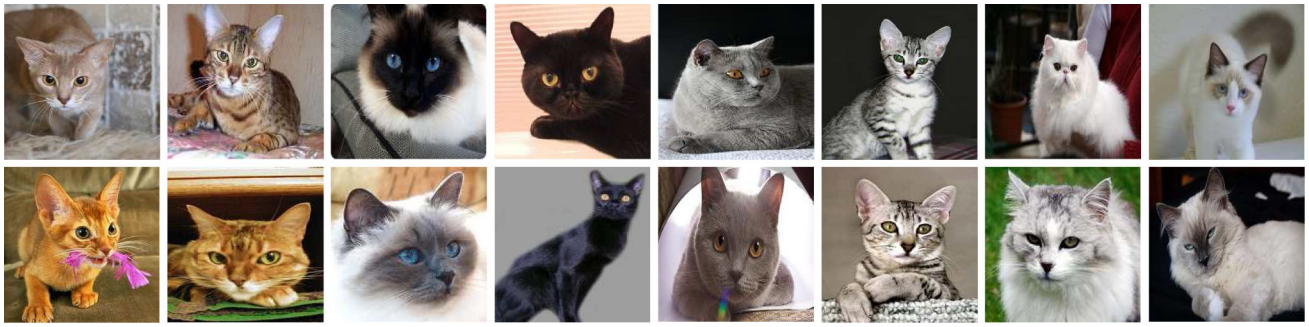

Fig. 1 Sample cat images from the Oxford-IIIT Pet database. **a** Abyssinian, **b** Bengal, **c** Birman, **d** Bombay, **e** British Shorthair, **f** Egyptian Mau, **g** Persian, **h** Ragdoll

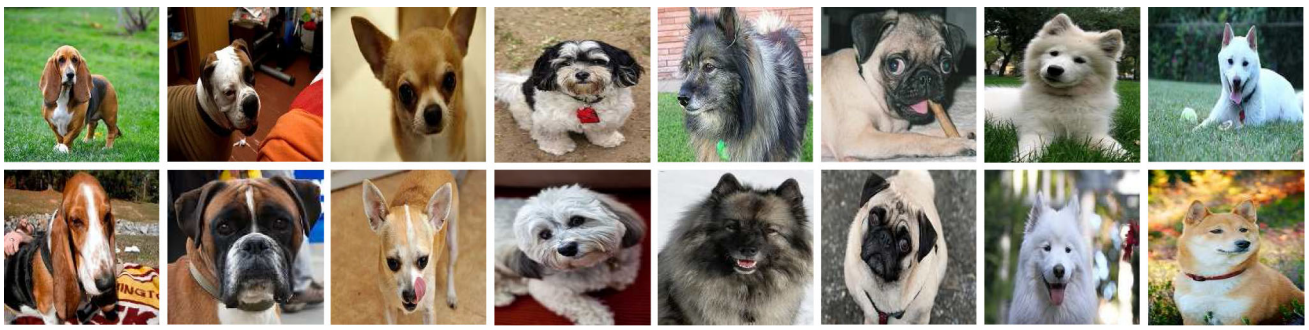

Fig. 2 Sample dog images from the Oxford-IIIT Pet database. **a** Basset hound, **b** Boxer, **c** Chihuahua, **d** Havanese, **e** Keeshond, **f** Pug, **g** Samyod, **h** Shiba inu

ture framework, and, therefore, the size of the vocabulary (i.e. the number of mixture components in the global-level mixture model) can be automatically inferred from the data thanks to the property of nonparametric Bayesian model. Then, the "bag-of-words" paradigm is employed and a histogram of "visual words" for each image is computed. Since the goal of our experiment is to determine which image category (breeds of cats and dogs) that a testing image $\mathcal{I}_j$ belongs to, we also need to introduce an indicator variable B_{jm} associated with each image (or group) in our hierarchical Dirichlet process mixture framework. B_{jm} denotes that image $\mathcal{I}_j$ comes from category m and is drawn from another infinite mixture model which is truncated at level J. This means that we need to add a new level of hierarchy to our hierarchical infinite mixture model with a sharing vocabulary among all image categories. In this experiment, we truncate J to 50 and initialize the hyperparameter of the mixing probability of B_{jm} to 0.05. Finally, we assign testing images to the appropriate categories according to Bayes' decision rule.

4.2.2 Data set

We conducted our experiments of categorizing cats and dogs using a publicly available database namely the Oxford-IIIT Pet database (Parkhi et al. 2012).[2] This database contains 7,349 images of cats and dogs in total with 12 different breeds of cats and 25 different breeds of dogs. Each of these breeds contains about 200 images. We randomly divided this database into two halves: one for training (to learn the model and build the visual vocabulary), the other one for testing. Sample images from the database are shown in Fig. 1 (cats) and Fig. 2 (dogs).

4.2.3 Results

In this experiment, our goal is to demonstrate the advantages of using hierarchical Dirichlet process framework over the conventional Dirichlet process one, as well as GD distribution over the Gaussian. Therefore, we compared the categorization results using the proposed *HInGD* model with three other mixture models including infinite GD mixture (*InGD*) model, hierarchical infinite Gaussian mixture (*HInGau*) model and infinite Gaussian mixture (*InGau*) model. All of these models were learned using variational inference. For the experiments of using *InGD* and *InGau* models, the visual vocabularies were built using the K-means algorithm

[2] Available at: http://www.robots.ox.ac.uk/~vgg/data/pets/.

Table 1 The average categorization performance (%) and the standard deviation obtained over 30 runs using different approaches

Method	Cats	Dogs
HInGD	54.72 (1.07)	42.93 (1.13)
InGD	51.25 (1.14)	40.47 (0.99)
HInGau	48.17 (0.96)	36.51 (0.82)
InGau	46.03 (1.08)	34.26 (1.23)

Table 2 The average categorization performance (%) and the standard deviation obtained over 30 runs using different approaches

Method	Cats and Dogs
HInGD	40.78 (1.19)
InGD	38.13 (1.15)
HInGau	35.39 (1.03)
InGau	34.15 (1.12)

and the sizes of the visual vocabularies were manually set to 500. We evaluated the categorization performance by running the approach 30 times.

First, we focus on discriminating different breeds of cats (12 breeds) and dogs (25 breeds), respectively. The average breed discrimination performances of our approach and the three other tested approaches are shown in Table 1. According to these results, the proposed *HInGD* approach was able to provide the highest categorization accuracies (54.72 % for cats and 42.93 % for dogs) among all tested approaches. The differences in accuracy between the *HInGD* and the other tested models are statistically significant according to Student's t test (i.e. for different runs, we have observed p values between 0.017 and 0.038 in the case of discriminating cats images, and p values between 0.022 and 0.041 for discriminating dogs images). Moreover, *HInGD* and *HInGau* outperformed *InGD* and *InGau*, respectively, which demonstrates the merits of using hierarchical Dirichlet process framework over the conventional Dirichlet process. Better modelling capabilities of using GD mixture models over Gaussian mixtures are also illustrated in Table 1, in terms of higher categorization accuracies obtained by *HInGD* and *InGD* than *HInGau* and *InGau*.

Next, we evaluated our approach and the other three approaches on discriminating different breeds of cats and dogs using the whole Oxford-IIIT Pet database (i.e. we do not separate cat and dog images). The corresponding results are summarized in Table 2. As we can see in this table, the proposed *HInGD* approach again provided higher categorization accuracy rate (40.78 %) than the other tested approaches (the differences are statistically significant as shown by a Student's t test, p values between 0.029 and 0.043).

4.3 Web service intrusion detection

4.3.1 Problem statement

Recent advances in web technologies and the constant growth of the Internet have led to many online service applications. Examples include e-commerce, social networks, online banking, business intelligence, web search engines, etc. An important feature of these web services is that they are based on software applications running at the server-side and generating new web content in an online fashion, which makes them flexible to exchange information on the Internet (Pearce et al. 2005; Desmet et al. 2005; Mehdi et al. 2012). The flexibility of web services poses also vulnerabilities which make them the targets for attacks (e.g. code injection attacks, SQL/XML injection, buffer overflow attacks, denial of service, etc.) by cyber-criminals who can collect confidential information from servers or even compromise them (Pinzen et al. 2010; Dagdee and Thakar 2008; Yee et al. 2007; Gruschka and Luttenberger 2006; Jensen et al. 2007). Then, there is an urgent need to protect the servers on which the applications are running (Zolotukhin et al. 2013; Zolotukhin and Hamalainen 2013; Jensen et al. 2009; Corona and Giacinto 2010). Indeed, intrusion detection systems (IDSs) need to be deployed. An overview of current intrusion detection techniques and related issues was proposed in Zhou et al. (2010) and Tsai et al. (2009). Recently, data mining and machine learning approaches have been used in this growing area to improve the performance of existing systems (Patcha and Park 2007; Laskov et al. 2005; Zanero and Savaresi 2004; Fan et al. 2011; Horng et al. 2011; Khan et al. 2007). The key idea for these works is to use machine learning techniques (e.g. decision trees, artificial neural networks, support vector machines, mixture models, etc.) to train a classifier and to recognize attacks based on a list of features, which generally reduces the intrusion detection problem to an adversarial learning task (Lowd and Meek 2005).

Based on the analysis methods, IDSs are usually classified into two main categories: misuse (i.e. signature-based) detection and anomaly detection systems (Northcutt and Novak 2002). In misuse detection systems, the goal is to detect the occurrence of attacks that have been previously identified as intrusions. For this type of IDS, attacks must be known a priori. Misuse detection can be viewed then as a supervised learning problem. Alternatively, anomaly detection systems detect unknown attacks by observing deviations from normal activities of the system. It is based on the assumption that intrusive activities are noticeably different from normal system activities and hence detectable. Data clustering and unsupervised learning approaches have been widely used to develop anomaly detection systems. Several of recent clustering approaches quantify deviation from normal behaviour using thresholds (see, for instance, Pereira and Jamhour

2013; Zolotukhin et al. 2013; Zolotukhin and Hamalainen 2013). Unlike these approaches we consider here our *HInGD* to model normal traffic data and then to automatically detect potential intrusions (i.e. anomalous traffic).

4.3.2 Results

The proposed framework is tested using logs collected from a real-life web service (from several Apache servers) in a two-week time interval. The collected data set contains normal requests, anomalies as well as intrusions. More specifically, our training data are collected at the beginning and is composed of 10,000 requests. The majority of these requests are legitimate, but some are attacks (e.g. cross-site scripting, SQL injections, buffer overflows, etc.). After using these data to train our mixture model, by considering 1 g, 2 g, and 3 g representations (an n-gram is a sub-sequence of n overlapping items from a given sequence) as done in Zolotukhin and Hamalainen (2013), new requests are considered and classified as normal or abnormal. More specifically, we performed our approach as a classifier to detect abnormal requests by assigning the testing request to the group (normal or abnormal) that most likely generated it. These new requests constitute the testing set and their number is equal to 35,000.

The evaluation of the performance of our approach has been based on the following measures:

- True-positive rate: the number of correctly detected intrusions over the number of intrusions in the testing set.
- False-positive rate: the number of normal requests considered as intrusions over the total number of normal requests in the testing set.
- True-negative rate: the number of correctly classified normal requests over the total number of normal requests in the testing set.
- False-negative rate: the number of misclassified intrusions over the number of intrusions in the testing set.
- Accuracy: the number of correctly classified requests over the total number of requests in the testing set.
- Precision: the number of correctly classified intrusions over the number of intrusions.

To demonstrate the advantages of our approach, we compared it with the SDEM and SDPU approaches in Yamanishi et al. (2004) based on Gaussian mixture models and kernel mixtures. Moreover, we performed comparisons with the well-known nearest-neighbour (KNN) technique and three recent state-of-the art approaches, namely GHSOMs (Zolotukhin et al. 2013), diffusion maps (Kirchner 2010), and the algorithm proposed in Zolotukhin and Hamalainen (2013), respectively. For the KNN approach, we have tested several values of K and found that the best performance was achieved when $K = 9$ according to the experimental results. For other tested approaches, we adopted the same initial experimental settings as in their original works. We evaluated the performance of each tested approach in detecting web service intrusion with 1 g representation and the corresponding results are illustrated in Table 3. According to this table, *HInGD* has provided the best performance among all tested approach in terms of the highest true-positive rate (98.71 %), the lowest false-positive rate (0.95 %), the highest true-negative rate (99.03 %), the lowest false-negative rate (0.99 %), the highest accuracy (98.26 %), and the highest precision (98.79 %). The fact that better performance provided by *HInGD* than the ones obtained by SDEM and SDPU implies that, the hierarchical Dirichlet process framework with GD mixtures works better than the SDEM algorithm with a finite Gaussian mixture and the SDPU algorithm with a kernel mixture in detecting web service intrusion. The performances achieved by SDEM and SDPU are comparable (the differences are not statistically significant, p values between 0.39 and 0.51). The difference is statistically significant when the *HInGD* is compared to SDEM and SDPU (p values between 0.037 and 0.048 for all evaluation measurers), respectively. Moreover, *HInGD* performed better than GHSOMs, and the approaches proposed in Kirchner (2010) and Zolotukhin and Hamalainen (2013) demonstrate the merits of using our approach over these state-of-the art approaches in web service intrusion detection. We may also observe that in Table 3, the KNN

Table 3 Performance (%) of using different approaches with 1 g representation in detecting web service intrusion

	HInGD	SDEM	SDPU	KNN	GHSOMs	Kirchner (2010)	Zolotukhin and Hamalainen (2013)
True-positive rate	98.71	97.90	97.91	95.02	98.01	98.00	98.04
False-positive rate	0.95	1.00	1.01	1.33	1.02	1.07	1.03
True-negative rate	99.03	98.56	98.50	94.14	98.60	98.55	98.65
False-negative rate	0.99	1.02	1.03	2.51	1.07	1.24	1.06
Accuracy	98.26	97.87	97.85	94.38	97.70	97.74	97.79
Precision	98.79	98.02	98.08	94.66	98.08	98.03	98.15

Table 4 Performance (%) of using different approaches with 2 g representation in detecting web service intrusion

	HInGD	SDEM	SDPU	KNN	GHSOMs	Kirchner (2010)	Zolotukhin and Hamalainen (2013)
True-positive rate	98.72	97.97	97.98	95.09	98.01	98.00	98.04
False-positive rate	0.94	1.01	1.02	2.22	1.02	1.06	1.03
True-negative rate	99.07	98.60	98.65	94.35	98.60	98.56	98.68
False-negative rate	0.97	1.02	1.03	2.43	1.04	1.24	1.10
Accuracy	98.32	97.92	97.90	94.56	97.76	97.74	97.79
Precision	98.87	98.11	98.13	94.73	98.09	98.08	98.18

Table 5 Performance (%) of using different approaches with 3 g representation in detecting web service intrusion

	HInGD	SDEM	SDPU	KNN	GHSOMs	Kirchner (2010)	Zolotukhin and Hamalainen (2013)
True-positive rate	98.79	98.09	98.08	95.15	98.08	98.03	98.05
False-positive rate	0.94	1.02	1.03	2.13	1.01	1.06	1.02
True-negative rate	99.11	98.65	98.66	94.41	98.65	98.63	98.68
False-negative rate	0.96	1.05	1.05	2.39	1.04	1.15	1.10
Accuracy	98.69	97.92	97.91	94.77	97.88	97.77	97.81
Precision	98.96	98.17	98.18	94.85	98.17	98.09	98.19

approach has provided the worst performance among all tested approaches. This can be explained by the fact that KNN approach has a suboptimal generalization power and is not robust to noisy data, also. Furthermore, we have tested the performance of our approach in detecting web service intrusion with 2-, and 3-g representation and the results are shown in Tables 4 and 5. According to these tables, we can see that the difference between the results obtained using the three representations is not important in our case. The results shown demonstrate also that our statistical framework is promising. Future works could be devoted to the analysis of the influence of the features representations on the results.

5 Conclusion

Our main goal in this paper, which we believe we have reached, was to develop a flexible statistical model for data modelling and classification. Our scheme is based on a powerful nonparametric Bayesian approach namely hierarchical Dirichlet process and a flexible probability density function namely the GD distribution, and a principled variational learning approach. The proposed framework has been shown to be statistically plausible, principled and well behaved and then can deal with many real-world problems. Indeed, it has provided state-of-the art results on two challenging applications namely visual objects categorization and web service intrusion detection. It is noteworthy that the proposed model is clearly scalable and appropriate for applications generating large scale data. We are currently investigating several future works related to the proposed model such as integrating feature selection to improve more its generalization capabilities, and extending the learning approach to online settings to take into account dynamic data.

Acknowledgments The second author would like to thank King Abdulaziz City for Science and Technology (KACST), Kingdom of Saudi Arabia, for their funding support under grant number 11-INF1787-08. The authors would like to thank the anonymous referees and the associate editor for their comments.

References

Agarwal S, Roth D (2002) Learning a sparse representation for object detection. In: Heyden A, Sparr G, Nielsen M, Johansen P (eds) ECCV (4), Lecture notes in computer science vol 2353. Springer, Berlin, Heidelberg, pp 113–130

Attias H (1999) A variational Bayes framework for graphical models. In: Proceedings of advances in neural information processing systems (NIPS), pp 209–215

Banerjee A, Merugu S, Dhillon IS, Ghosh J (2004) Clustering with bregman divergences. In: Proceedings of the 4th SIAM international conference on data mining (SDM), pp 234–245

Bishop CM (2006) Pattern recognition and machine learning. Springer, New York

Blei DM, Jordan MI (2005) Variational inference for Dirichlet process mixtures. Bayesian Anal 1:121–144

Bouguila N, Ziou D (2005) Using unsupervised learning of a finite dirichlet mixture model to improve pattern recognition applications. Pattern Recognit Lett 26(12):1916–1925

Bouguila N, Ziou D (2006) A hybrid SEM algorithm for high-dimensional unsupervised learning using a finite generalized Dirichlet mixture. IEEE Trans Image Process 15(9):2657–2668

Bouguila N, Ziou D (2007) High-dimensional unsupervised selection and estimation of a finite generalized Dirichlet mixture model based on minimum message length. IEEE Trans Pattern Anal Mach Intell 29(10):1716–1731

Bouguila N, Ziou D (2010) A dirichlet process mixture of generalized dirichlet distributions for proportional data modeling. IEEE Trans Neural Netw 21(1):107–122

Boutemedjet S, Bouguila N, Ziou D (2009) A hybrid feature extraction selection approach for high-dimensional non-Gaussian data clustering. IEEE Trans Pattern Anal Mach Intell 31(8):1429–1443

Corona I, Giacinto G (2010) Detection of server-side web attacks. In: Diethe T, Cristianini N, Shawe-Taylor J (eds) JMLR Proceedings, WAPA, vol 11, JMLR.org, pp 160–166

Dagdee N, Thakar U (2008) Intrusion attack pattern analysis and signature extraction for web services using honeypots. In: Proceedings of the First international conference on emerging trends in engineering and technology (ICETET), p 1232–1237

Desmet L, Jacobs B, Piessens F, Joosen W (2005) Threat modelling for web services based web applications. In: Chadwick D, Preneel B (eds) Communications and multimedia security, vol 175. IFIPG The International Federation for Information ProcessingSpringer, US, pp 131–144

Fan W, Bouguila N, Ziou D (2011) Unsupervised anomaly intrusion detection via localized bayesian feature selection. In: Proceedings of the EEE international conference on data mining (ICDM), pp 1032–1037

Fan W, Bouguila N (2013) Variational learning of a Dirichlet process of generalized Dirichlet distributions for simultaneous clustering and feature selection. Pattern Recognit 46(10):2754–2769

Fan W, Bouguila N, Ziou D (2013) Unsupervised hybrid feature extraction selection for high-dimensional non-gaussian data clustering with variational inference. IEEE Transa Knowl Data Eng 25(7):1670–1685

Ferguson TS (1983) Bayesian density estimation by mixtures of normal distributions. Recent Adv Stat 24:287–302

Gruschka N, Luttenberger N (2006) Protecting web services from dos attacks by soap message validation. In: Fischer-Hebner S, Rannenberg K, Yngstram L, Lindskog S (eds) Security and privacy in dynamic environments, vol 201. IFIP International Federation for Information ProcessingSpringer, US, pp 171–182

Horng S-J, Su M-Y, Chen Y-H, Kao T-W, Chen R-J, Lai J-L, Perkasa CD (2011) A novel intrusion detection system based on hierarchical clustering and support vector machines. Expert Syst Appl 38(1):306–313

Ishwaran H, James LF (2001) Gibbs sampling methods for stick-breaking priors. J Am Statistical Assoc 96:161–173

Jain AK, Topchy A, Law MHC, Buhmann JM (2004) Landscape of clustering algorithms. In: Proceedings of the 17th international conference on pattern recognition (ICPR), vol 1. pp 260–263

Jensen M, Gruschka N, Herkenhener R (2009) A survey of attacks on web services. Comput Sci Res Dev 24(4):185–197

Jensen M, Gruschka N, Herkenhoner R, Luttenberger N (2007) Soa and web services: new technologies, new standards—new attacks. In: Proceedings of the fifth European conference on web services (ECOWS), pp 35–44

Kahn JM (2004) A generative bayesian model for aggregating experts' probabilities. In: Proceedings of the 20th conference in uncertainty in artificial intelligence (UAI), AUAI Press, pp 301–308

Ke Y, Sukthankar R (2004) PCA-SIFT: A more distinctive representation for local image descriptors. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), pp 506–513

Khan L, Awad M, Thuraisingham B (2007) A new intrusion detection system using support vector machines and hierarchical clustering. VLDB J 16(4):507–521

Kirchner M (2010) A framework for detecting anomalies in http traffic using instance-based learning and k-nearest neighbor classification. In: Proceedings of the 2nd international workshop on security and communication networks (IWSCN), pp 1–8

Korwar RM, Hollander M (1973) Contributions to the theory of dirichlet processes. Ann Probab 1:705–711

Lamdan Y, Schwartz JT, Wolfson HJ (1988) Object recognition by affine invariant matching. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), pp 335–344

Laskov P, Dessel P, Schefer C, Rieck K (2005) Learning intrusion detection: supervised or unsupervised? In: Roli F, Vitulano S (eds) Image analysis and processing (ICIAP), Lecture notes in computer science vol 3617. Springer, Berlin, pp 50–57

Law MHC, Topchy AP, Jain AK (2005) Model-based clustering with probabilistic constraints. In: Proceedings of the SIAM international conference on data mining (SDM), pp 641–645

Lazebnik S, Schmid C, Ponce J (2004) Semi-local affine parts for object recognition. In: Proceedings of the British machine vision conference (BMVC), BMVA Press, pp 1–10

Li B, Zhong R-T, Wang X-J, Zhuang Z-Q (2006) Continuous optimization based-on boosting gaussian mixture model. In: Proceedings of the 18th international conference on pattern recognition (ICPR), vol 1. pp 1192–1195

Lowd D, Meek C (2005) Adversarial learning. In: Proceedings of the Eleventh ACM SIGKDD international conference on knowledge discovery and data mining (KDD), pp 641–647

Lu Q, Yao X (2005) Clustering and learning gaussian distribution for continuous optimization. IEEE Trans Syst Man Cybern Part C Appl Rev 35(2):195–204

Matas J, Koubaroulis D, Kittler J (2002) The multimodal neighborhood signature for modeling object color appearance and applications in object recognition and image retrieval. Comput Vis Image Underst 88(1):1–23

McLachlan GJ, Peel D (2000) Finite mixture models. Wiley, New York

Mehdi M, Bouguila N, Bentahar J (2012) Trustworthy web service selection using probabilistic models. In: Proceedings of the IEEE 19th international conference on web services (ICWS), pp 17–24

Mikolajczyk K, Schmid C (2004) Scale and affine invariant interest point detectors. Int J Comput Vis 60:63–86

Northcutt S, Novak J (2002) Network intrusion detection: an analyst's handbook. New Riders Publishing, UK

Parkhi OM, Vedaldi A, Zisserman A, Jawahar CV (2013) Cats and dogs. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), pp 3498–3505

Patcha A, Park J-M (2007) An overview of anomaly detection techniques: existing solutions and latest technological trends. Comput Netw 51(12):3448–3470

Pearce C, Bertok P, Schyndel R (2005) Protecting consumer data in composite web services. In: Sasaki R, Qing S, Okamoto E, Yoshiura H (eds) Security and privacy in the age of ubiquitous computing, vol 181. IFIP Advances in Information and Communication Technology Springer, US, pp 19–34

Pereira H, Jamhour E (2013) A clustering-based method for intrusion detection in web servers. In: Proceedings of the 20th international conference on telecommunications (ICT), pp 1–5

Pinzen C, Paz JF, Zato C, Perez J (2010) Protecting web services against dos attacks: A case-based reasoning approach. In: Romay M, Corchado E, Garcia Sebastian MT (eds) Hybrid artificial intelligence systems, Lecture notes in computer science, vol 6076. Springer, Berlin, pp 229–236

Rasiwasia N, Vasconcelos N (2008) Scene classification with low-dimensional semantic spaces and weak supervision. In: Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), p 1–6

Sethuraman J (1994) A constructive definition of Dirichlet priors. Statistica Sin 4:639–650

Shoham S, Fellows MR, Normann RA (2003) Robust, automatic spike sorting using mixtures of multivariate t-distributions. J Neurosci Methods 127(2):111–122

Teh Y-W, Jordan MI, Beal MJ, Blei DM (2006) Hierarchical Dirichlet processes. J Am Stat Assoc 101(476):1566–1581

Teh YW, Jordan MI (2010) Hierarchical Bayesian nonparametric models with applications. In: Hjort N, Holmes C, Müller P, Walker S (eds) Bayesian nonparametrics: principles and practice. Cambridge University Press, London

Tsai C-F, Hsu Y-F, Lin C-Y, Lin W-Y (2009) Review: intrusion detection by machine learning: a review. Expert syst Appl 36(10):11994–12000

Wang C, Paisley JW, Blei DM (2011) Online variational inference for the hierarchical Dirichlet process. J Mach Learn Res Proc Track 15:752–760

Xiang S, Nie F, Zhang C (2008) Learning a mahalanobis distance metric for data clustering and classification. Pattern Recognit 41(12):3600–3612

Yamanishi K, Takeuchi J-I, Williams GJ, Milne P (2004) On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms. Data Min Knowl Discov 8(3):275–300

Yee CG, Shin WH, Rao G (2007) An adaptive intrusion detection and prevention (ID/IP) framework for web services. In: Proceedings of the international conference on convergence information technology (ICCIT), p 528–534

Zanero S, Savaresi SM (2004) Unsupervised learning techniques for an intrusion detection system. In: Proceedings of the ACM symposium on applied computing (SAC), ACM, pp 412–419

Zhou CV, Leckie C, Karunasekera S (2010) A survey of coordinated attacks and collaborative intrusion detection. Comput Secur 29(1):124–140

Zolotukhin M, Hamalainen T (2013) Detection of anomalous http requests based on advanced n-gram model and clustering techniques. In: Balandin S, Andreev S, Koucheryavy Y (eds) Internet of things., smart spaces, and next generation networking, Lecture notes in computer science, vol 8121. Springer, Berlin, pp 371–382

Zolotukhin M, Hamalainen T, Juvonen A (2013) Growing hierarchical self-organizing maps and statistical distribution models for online detection of web attacks. In: Cordeiro J, Krempels KH (eds) Web information systems and technologies, Lecture notes in business information processing vol 140. Springer, Berlin, pp 281–295