SCIENCE CHINA
Information Sciences

March 2021, Vol. 64 130104:1–130104:11
https://doi.org/10.1007/s11432-020-3118-7

• RESEARCH PAPER •
Special Focus on Weakly Supervised Learning

Crowdsourcing aggregation with deep Bayesian learning

Shao-Yuan LI[1*], Sheng-Jun HUANG[1,2] & Songcan CHEN[1]

[1]*College of Computer Science and Technology, College of Artificial Intelligence, Nanjing University of Aeronautics and Astronautics, Nanjing* 211106*, China;*
[2]*State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing* 210093 *China*

Received 30 June 2020/Accepted 30 July 2020/Published online 7 February 2021

Abstract In this study, we consider a crowdsourcing classification problem in which labeling information from crowds is aggregated to infer latent true labels. We propose a fully Bayesian deep generative crowdsourcing model (BayesDGC), which combines the strength of deep neural networks (DNNs) on automatic representation learning and the interpretable probabilistic structure encoding of probabilistic graphical models. The model comprises a DNN classifier as a prior for the true labels and a probabilistic model for the annotation generation process. The DNN classifier and annotation generation process share the latent true label variables. To address the inference challenge, we developed a natural-gradient stochastic variational inference, which combines variational message passing for conjugate parameters and stochastic gradient descent for DNN and learns the distribution of latent true labels and workers' confusion matrix via end-to-end training. We illustrated the effectiveness of the proposed model using empirical results on 22 real-world datasets.

Keywords crowdsourcing, classification, fully Bayesian deep generative models, natural gradient, stochastic variational inference

Citation Li S-Y, Huang S-J, Chen S C. Crowdsourcing aggregation with deep Bayesian learning. Sci China Inf Sci, 2021, 64(3): 130104, https://doi.org/10.1007/s11432-020-3118-7

1 Introduction

Typical supervised learning requires training labels. However, for many real-world tasks, acquiring the gold standard in terms of labeling is not possible or too expensive. In recent years, however, crowdsourcing [1, 2] has been established as a reliable solution for collecting data annotations. With the advent of crowdsourcing services such as Amazon Mechanical Turk[1)] and Crowdflower[2)], crowdsourcing has been used for collecting vast annotated datasets in a time period in numerous fields such as natural language understanding [3], medical diagnosis [4], vision image tagging [5], and entity resolution [6].

Although crowdsourcing is sufficiently scalable, the annotations provided by annotators are inherently subjective and there can be a substantial degree of disagreement among different annotators. The noise associated with annotations can lead to limiting the performance of conventional learning algorithms. Consequently, a core task in crowdsourcing is estimating the hidden ground truth labels from collected annotations. Several methods have been proposed for this purpose. In this study, these methods are based on whether they use only the annotation information or data feature information and, as such, are categorized into two groups. Among the methods based only on annotation information, majority voting is the simplest and most common technique, which treats annotators as equally reliable and grants workers equal votes. To consider the annotators' skills and data difficulty variation, probabilistic models have been developed. As an early study in this context, Dawid and Skene [7] parameterized annotators' reliability using their error rates and modeled the annotations as noisy observations of the latent ground truth.

* Corresponding author (email: lisy@nuaa.edu.cn)
1) https://www.mturk.com.
2) http://crowdflower.com.

By extending [7] with advanced levels of annotation generation processes and optimization techniques, additional studies were conducted [8–10].

Because the data features include complementary information, different approach proceeds by integrating these features into a learning model [4, 11–15]. Work conducted by Raykar et al. [4] serves as among the most prominent in this regard and extends [7] to enable joint learning of the worker parameters and a logistic regression classifier. Treating the classifier as a prior of the ground truth label, the optimization naturally follows the expectation-maximization (EM) procedure of [7]. This idea was later extended to other types of classifier models such as Gaussian process classifiers [11] and recently to deep neural networks, known as deep crowd learning (DCL). The studies of [12, 14, 15] are three examples of DCL. Ref. [12] exploited a convolutional neural network as a classifier and used the EM optimization procedure. To avoid the computational overhead of iterative EM in [12], Refs. [14, 15] distinguished themselves from [12] by proposing treatment of the latent true labels as a single hidden layer of the deep neural network (DNN) and adopting crowd annotations as the output layer. The entire DNN was directly trained end-to-end using noisy labels and backpropagation.

While [14, 15] combated the computational issue of EM-style algorithms, their heuristic optimization implementation cannot guarantee the maximization of specific lower bounds of the original learning objective, unlike EM-based algorithms. Moreover, they lose the probabilistic structure interpretation of the DNN classifier output and worker parameterization. In this study, to retain the strength of the DNN in automatic representation learning and the flexibility of probabilistic graphical models for encoding interpretable probabilistic structures, we propose a fully Bayesian deep generative crowdsourcing model (BayesDGC).

In particular, we exploit the principle idea of considering the DNN classifier as a prior for the true labels and parameterize each worker's reliability using a confusion matrix. The latent true label variables are shared among the DNN classifier and the annotation generation process. Unlike the work of [14, 15], which heuristically learned uninterpretable deterministic parameters and required human tuning, our model was fully Bayesian. The BayesDGC performs distribution inference for the latent true labels and the worker's confusion matrix, thus automatically providing a trade-off between the model complexity and data fitting. To address the inference challenge, we developed a natural-gradient stochastic variational inference algorithm that combined variational message passing for conjugate structures and the SGD (stochastic gradient descent) of the DNN and conducted all parameter training in an end-to-end manner using backpropagation. The optimization process guaranteed to maximizing a variational lower bound of the observed annotations' likelihood.

In this study, while the modeling of the independent worker confusion matrix is basic, we note that the proposed deep Bayesian inference is sufficiently general for application to more sophisticated parameterizations, provided that the parameter conjugate structures can be exploited. In future, we aim to adopt deep Bayesian crowdsourcing involving more sophisticated annotation generation process such as correlated workers [10, 16].

2 Related work

In the last few years, several methods have been proposed for crowd aggregation to address annotation noise and trustworthiness issues. Among them, majority voting (MV) is the most straightforward and extensively used and conducts simple voting involving all workers. Because MV ignores quality differences in worker annotations, [17] and [18], respectively, proposed strategies considering certainty information of the majority and minority classes, as well as quality differences of workers over different instances. Probabilistic approaches modeling the workers' expertise and instances' difficulties are another exploration line. The DS (Dawid & Skene) model [7] is a key early contribution in this regard. To address the clinical diagnostics problem, the DS model proposes using error rates to parameterize worker labels as conditioned on the item's true label, and proposes an EM algorithm to estimate error rates and latent true labels. The generative annotation modeling idea has served as a basis for many other variants that model the annotation generation process in more detailed levels using different optimization techniques. For example, Refs. [5, 8] considered item difficulty and proposed an EM algorithm to infer the most probable label. Ref. [9] used a confusion matrix for each item and estimated the latent true labels via a minimax entropy principle, thus promoting the true label distribution close to empirical worker annotation distributions. Ref. [19] conducted optimization in crowdsourcing from a variational inference perspective

and proposed variational inference methods including belief propagation and mean-field models. Bayesian extensions of DS were explored in [20–22], which generalized DS to being fully Bayesian by introducing Dirichlet priors and conducting inference, respectively, through Gibbs sampling, variational Bayesian inference, and EM. Recently, rather than independently treating the workers, modeling correlations between workers has attracted considerable attention. In [23], a non-parametric Dirichlet process is used to explicitly model workers in clusters within which confusion matrices are possibly similar. Ref. [16] derived a minimax error rate for general confusion-matrix-based models and proposed a worker-clustering model. Ref. [10] proposed a mixture model for classes and created links between worker correlation and annotation tensor decomposition.

Rather than relying purely on annotations to infer the truth, work on using data feature information to help improve true label estimation have been conducted. Ref. [4] was one of the pioneers in this area by extending DS using a logistic regression classifier as the true label prior. Moreover, other types of classifier models such as a Gaussian process classifier and supervised latent Dirichlet allocation [11, 24] were proposed. These methods primarily work by integrating a supervised learning model as a prior of true labels and adding it to the probabilistic annotation generation model. Ref. [25] proposed constructing local linear neighborhood graph in the feature space and conducting annotation distribution propagation in the label space.

With the success of DNN that allow for flexible data representations to be learned [26], deep crowd learning attempting to combine DNN with crowdsourcing was performed [12–15, 27]. Ref. [12] used a convolutional neural network (CNN) classifier as a label prior and applied the EM optimization procedure. Refs. [14, 15] avoided the computational overhead of the EM by heuristically conducting direct loss minimization on noisy annotations, thereby applying DNN SGD optimization. Technically, our work is inspired by [13, 27] who exploited deep generative models and conducted the inference using end-to-end backpropagation. Our study differs in terms of problem scenario and implementation. Ref. [13] considered semi-supervised crowd classification based on the inference technique of a non-Bayesian semi-supervised variational autoencoder [28]. However, Ref. [27] considered the clustering problem. Accordingly, model construction and inference implementations are entirely different.

3 The proposed model

We denote the set of N examples' observations by $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ shows the d-dimensional feature values of the i-th example. The collected annotations provided by W workers are denoted as $\boldsymbol{L} \in \{0, 1, \ldots, K\}^{N \times W}$, with $\boldsymbol{L}_{ij}$ representing the label assignment of example i given by the worker j. When $\boldsymbol{L}_{ij} = k$ $(k \neq 0)$, the i-th example is categorized as a k-th class by the j-th worker. When $\boldsymbol{L}_{ij} = 0$, it indicates that the annotations of worker j for example i is not observed. Our target is to estimate the latent true labels $\boldsymbol{Y} = \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N\}$ for $\boldsymbol{X}$, thereby making the best use of feature and crowd annotations.

3.1 Fully Bayesian deep generative crowdsourcing (BayesDGC)

Figure 1 shows the graphical model of the proposed approach. The model comprises two main parts: the annotation generation process $p(\boldsymbol{L}_{ij}|\boldsymbol{y}_i; \boldsymbol{V}_j)$ and the prior model for latent true labels $p(\boldsymbol{y}_i|\boldsymbol{x}_i, \pi)$.

For the annotation generation part, we adopted classic independent confusion matrix parameterization for each worker. In particular, the confusion matrix of the j-th worker can be characterized by parameters $\boldsymbol{V}_j = \{\boldsymbol{\nu}_{j1}, \ldots, \boldsymbol{\nu}_{jK}\}$, with vector $\boldsymbol{\nu}_{jk} = \{\nu_{jk1}, \ldots, \nu_{jkK}\}$. Given the true label $\boldsymbol{y}_i$ of one example $\boldsymbol{x}_i$, the generation likelihood of annotation $\boldsymbol{L}_{ij}$ is as follows:

$$p(\boldsymbol{L}_{ij} = l|\boldsymbol{y}_i = k, \boldsymbol{V}_j) = \nu_{jkl}. \tag{1}$$

Assuming the examples are independent and the annotations for each example are independently generated by different workers, the total likelihood of the annotations can be written as

$$P(\boldsymbol{L}|\boldsymbol{Y}, \boldsymbol{V}) = \prod_{i=1}^{N} \prod_{j=1}^{W} \mathcal{I}[\boldsymbol{L}_{ij} \neq 0] p(\boldsymbol{L}_{ij}|\boldsymbol{y}_i, \boldsymbol{V}_j). \tag{2}$$

For the latent true labels' prior model, we exploit two priors, i.e., one data invariant prior $p(\boldsymbol{y}_i; \boldsymbol{\pi})$ and one feature dependent neural network classifier prior $p(\boldsymbol{y}_i|\boldsymbol{x}_i; \boldsymbol{\gamma})$ parameterized by $\boldsymbol{\gamma}$, respectively, and

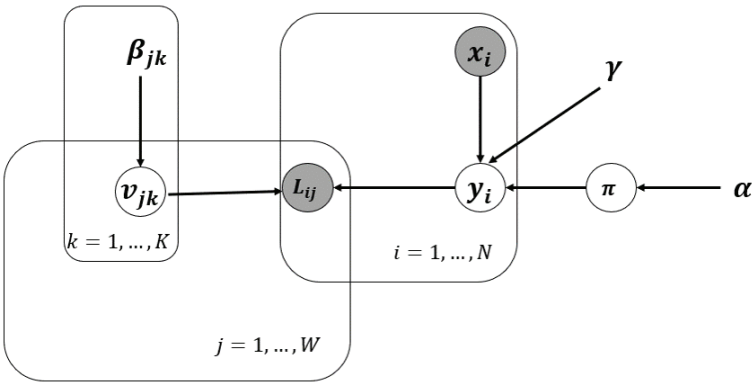

Figure 1 The plate notation for our proposed BayesDGC.

defined as follows:

$$p(\boldsymbol{y}_i, \boldsymbol{\pi}) = \prod_{k=1}^{K} \boldsymbol{\pi}_k, \quad p(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\gamma}) = \text{Categorical}(\boldsymbol{\tau}(\boldsymbol{x}_i; \boldsymbol{\gamma})). \tag{3}$$

Assuming that the examples are independent, the prior of $\boldsymbol{Y}$ can be written as follows:

$$p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\pi}, \boldsymbol{\gamma}) = p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\gamma}) p(\boldsymbol{Y}|\boldsymbol{\pi}) = \prod_{i=1}^{N} p(\boldsymbol{y}_i, \boldsymbol{\pi}) p(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\gamma}). \tag{4}$$

In addition to the above annotation generation process and true label prior, we assume conjugate Dirichlet priors over the global parameters $\Theta = \{\boldsymbol{V}, \boldsymbol{\pi}\}$, which are defined as follows:

$$p(\boldsymbol{\pi}) = \text{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha}), \qquad p(\boldsymbol{V}) = \prod_{j=1}^{W} \prod_{k=1}^{K} p(\boldsymbol{\nu}_{jk}) = \text{Dir}(\boldsymbol{\nu}_{jk}|\boldsymbol{\beta}_{jk}). \tag{5}$$

Thus the overall joint distribution of the observed annotations $\boldsymbol{L}$, the latent true labels $\boldsymbol{Y}$, and global parameters $\boldsymbol{\Theta} = \{\boldsymbol{V}, \boldsymbol{\pi}\}$ can be represented as follows:

$$p(\boldsymbol{L}, \boldsymbol{Y}, \boldsymbol{\Theta}|\boldsymbol{X}, \boldsymbol{\gamma}) = p(\boldsymbol{\pi}) p(\boldsymbol{Y}|\boldsymbol{\pi}) p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\gamma}) p(\boldsymbol{L}|\boldsymbol{Y}, \boldsymbol{V}) p(\boldsymbol{V}). \tag{6}$$

Our aim is to estimate the posterior distribution of global parameters $p(\boldsymbol{\Theta}|\boldsymbol{L}, \boldsymbol{X})$ and true labels $p(\boldsymbol{Y}|\boldsymbol{L}, \boldsymbol{X})$, by maximizing the likelihood of observed annotations $p(\boldsymbol{L})$.

In the case without the DNN classifier prior, our model degenerated to the Bayesian extension to the DS model, which was independently implemented using optimization procedures such as Gibbs sampling [20], mean-field variational Bayes [21], and EM [22]. However, in our deep crowd model, when combined with nonlinear DNN classifier $p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\gamma})$, Gibbs sampling and EM are too slow, because they involved expensive sampling loops for data or an iterative procedure per epoch. Variational mean-field message passing is quite efficient, but it depends on conjugate exponential family likelihood to preserve tractable structures, which does not hold for general data models such as neural networks. In the next subsection, we build on recent advances in structured variational autoencoders (SVAE), and develop a stochastic variational inference algorithm for our model, that can conduct efficient end-to-end training of all parameters and guaranteed to maximize selected variational lower bounds of the log-likelihood of annotations $\log p(\boldsymbol{L})$.

3.2 Natural-gradient stochastic variational inference algorithm

To perform efficient inference in deep probabilistic graphical models, the variational autoencoder (VAE) [29] used a reparameterization technique and proposed using a recognition network to fit mapping from the data to the distribution parameters involved. Thus, the posterior distribution can be inferred through end-to-end optimization over the entire neural network. Using structured VAE (SVAE) [30], the authors extended VAE using the notion of natural gradient stochastic variation inference (SVI) [31] for

conditional conjugate models. This idea is straightforward, rather than using the recognition network to output the posterior distribution's parameters, the authors used the recognition network to output the conjugate graphical model potentials, which were then used for mean-field variational message passing and natural gradient computation. The advantage of SVAE is that, it can leverage a conjugate structure to efficiently compute natural gradients of variational parameters, which enables effective second-order optimization.

In this study, we follow the optimization procedure of SVAE and implement the natural gradient stochastic variational inference algorithm for our BayesDGC. In the following, we provide the details of this implementation. As in VAE [29], the variational evidence lower bound (ELBO) is derived as follows:

$$\log p(\boldsymbol{L}) \geqslant \mathcal{L}(\boldsymbol{Y}, \boldsymbol{\Theta}, \boldsymbol{\gamma}) \triangleq \mathbb{E}_{q(\boldsymbol{\Theta}, \boldsymbol{Y})}\left[\log \frac{p(\boldsymbol{L}, \boldsymbol{Y}, \boldsymbol{\Theta}|\boldsymbol{X}, \boldsymbol{\gamma})}{q(\boldsymbol{Y})q(\boldsymbol{\Theta})}\right]. \tag{7}$$

Here, we exploit a mean-field variational family, i.e., $q(\boldsymbol{\Theta}, \boldsymbol{Y}) = q(\boldsymbol{\Theta})q(\boldsymbol{Y})$. To use the conjugate structure of our model, we rewrite the distribution of $p(\boldsymbol{\pi})$, $p(\boldsymbol{V})$, $p(\boldsymbol{Y}|\boldsymbol{\pi})$ defined in Eqs. (4) and (5), in their exponential family form:

$$p(\boldsymbol{\pi}) = \exp\{\langle \boldsymbol{\eta}_{\boldsymbol{\pi}}, \boldsymbol{t}(\boldsymbol{\pi})\rangle - \log Z(\boldsymbol{\eta}_{\boldsymbol{\pi}})\}, \tag{8}$$

$$p(\boldsymbol{\nu}_{jk}) = \exp\{\langle \boldsymbol{\eta}_{\boldsymbol{\nu}_{jk}}, \boldsymbol{t}(\boldsymbol{\nu}_{jk})\rangle - \log Z(\boldsymbol{\eta}_{\boldsymbol{\nu}_{jk}})\}, \tag{9}$$

$$p(\boldsymbol{y}|\boldsymbol{\pi}) = \exp\{\langle \boldsymbol{\eta}_{\boldsymbol{y}}(\boldsymbol{\pi}), \boldsymbol{t}(\boldsymbol{y})\rangle - \log Z(\boldsymbol{\eta}_{\boldsymbol{y}}(\boldsymbol{\pi}))\} = \exp\{\langle \boldsymbol{t}(\boldsymbol{\pi}), (\boldsymbol{t}(\boldsymbol{y}), \boldsymbol{1})\rangle\}. \tag{10}$$

Here $\boldsymbol{\eta}$ denotes the natural parameters, $\boldsymbol{t}(\cdot)$ denotes the sufficient statistics, and $\log Z(\cdot)$ denotes the log partition function. For Eqs. (8)–(10), their expressions are as follows:

$$\boldsymbol{\eta}_{\boldsymbol{\pi}} = \begin{bmatrix} \boldsymbol{\alpha}_1 - 1 \\ \vdots \\ \boldsymbol{\alpha}_K - 1 \end{bmatrix}, \quad \boldsymbol{\eta}_{\boldsymbol{\nu}_{jk}} = \begin{bmatrix} \boldsymbol{\beta}_{jk1} - 1 \\ \vdots \\ \boldsymbol{\beta}_{jkK} - 1 \end{bmatrix}, \quad \boldsymbol{\eta}_{\boldsymbol{y}}(\boldsymbol{\pi}) = \begin{bmatrix} \log \boldsymbol{\pi}_1 \\ \vdots \\ \log \boldsymbol{\pi}_K \end{bmatrix},$$

$$\boldsymbol{t}(\boldsymbol{\pi}) = \begin{bmatrix} \log \boldsymbol{\pi}_1 \\ \vdots \\ \log \boldsymbol{\pi}_K \end{bmatrix}, \quad \boldsymbol{t}(\boldsymbol{\nu}_{jk}) = \begin{bmatrix} \log \boldsymbol{\nu}_{jk1} \\ \vdots \\ \log \boldsymbol{\nu}_{jkK} \end{bmatrix}, \quad \boldsymbol{t}(\boldsymbol{y}) = \begin{bmatrix} \boldsymbol{y}_1 \\ \vdots \\ \boldsymbol{y}_K \end{bmatrix},$$

$$\log Z(\boldsymbol{\eta}_{\boldsymbol{\pi}}) = \sum_{k=1}^{K} \log \Gamma(\boldsymbol{\alpha}_k) - \log \Gamma\left(\sum_{k=1}^{K} \boldsymbol{\alpha}_k\right),$$

$$\log Z(\boldsymbol{\eta}_{\boldsymbol{\nu}_{jk}}) = \sum_{l=1}^{K} \log \Gamma(\boldsymbol{\beta}_{jkl}) - \log \Gamma\left(\sum_{k=1}^{K} \boldsymbol{\beta}_{jkl}\right),$$

$$\log Z(\boldsymbol{\eta}_{\boldsymbol{y}}(\boldsymbol{\pi})) = 0.$$

Here, Γ is the Gamma function. Similarly, rewriting the posterior distribution in its exponential family form $q(\theta) = \exp\{\langle \boldsymbol{\eta}_\theta, \boldsymbol{t}(\theta)\rangle - \log Z(\theta)\}$, $\theta \in \boldsymbol{\Theta} \cup \boldsymbol{Y}$. Substituting the above mentioned exponential family expressions for distributions, the ELBO $\mathcal{L}(\boldsymbol{Y}, \boldsymbol{\Theta}; \boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ in Eq. (7) becomes $\mathcal{L}(\boldsymbol{\eta}_{\boldsymbol{Y}}, \boldsymbol{\eta}_{\boldsymbol{\Theta}}; \boldsymbol{\gamma}, \boldsymbol{\alpha}, \boldsymbol{\beta})$, with $\boldsymbol{\eta}$ as parameters:

$$\mathcal{L}(\boldsymbol{\eta}_{\boldsymbol{Y}}, \boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma}) \triangleq \mathbb{E}_{q(\boldsymbol{\Theta}, \boldsymbol{Y})}\left[\log \frac{p(\boldsymbol{L}, \boldsymbol{Y}, \boldsymbol{\Theta}|\boldsymbol{X}, \boldsymbol{\gamma})}{q(\boldsymbol{Y})q(\boldsymbol{\Theta})}\right]. \tag{11}$$

To leverage the conjugate structure of our model, as in SVAE [30], we use the DNN classifier to create conjugate graphical model potentials:

$$\psi(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\gamma}) \triangleq \langle \boldsymbol{\gamma}(\boldsymbol{x}_i), \boldsymbol{t}(\boldsymbol{y}_i)\rangle. \tag{12}$$

Replacing $p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\gamma})$ with the conjugate term defined by $\psi(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{\gamma})$, we get the following surrogate objective $\hat{\mathcal{L}}$:

$$\hat{\mathcal{L}}(\boldsymbol{\eta}_{\boldsymbol{Y}}, \boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma}) \triangleq \mathbb{E}_{q(\boldsymbol{\Theta}, \boldsymbol{Y})}\left[\log \frac{p(\boldsymbol{L}, \boldsymbol{Y}, \boldsymbol{\Theta}) \exp\{\psi(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\gamma})\}}{q(\boldsymbol{Y})q(\boldsymbol{\Theta})}\right]. \tag{13}$$

Then, similar to SVI [31], we can conduct the natural gradient stochastic variational inference. With the global variational parameters $\boldsymbol{\eta}_{\boldsymbol{\Theta}}$ fixed, the optimal solution for $q^*(\boldsymbol{Y})$ factorizes over examples, $q^*(\boldsymbol{Y}) = \prod_{i=1}^{N} q^*(\boldsymbol{y}_i)$. Each $q^*(\boldsymbol{y}_i)$ is then derived in the closed form:

$$\begin{aligned} &\log q^*(\boldsymbol{y}_i) = \mathbb{E}_{q(\boldsymbol{\pi})} \log p(\boldsymbol{y}_i|\boldsymbol{\pi}) + \langle \boldsymbol{\gamma}(\boldsymbol{x}_i), \boldsymbol{t}(\boldsymbol{y}_i) \rangle + \mathbb{E}_{q(\boldsymbol{V})} \log p(\boldsymbol{L}|\boldsymbol{Y}, \boldsymbol{V}) + \text{const}, \\ &\boldsymbol{\eta}^*_{\boldsymbol{y}_i} = \mathbb{E}_{q(\boldsymbol{\pi})} \boldsymbol{t}(\boldsymbol{\pi}) + \boldsymbol{\gamma}(\boldsymbol{x}_i) + \sum_{i=1}^{N} \sum_{j=1}^{W} \mathcal{I}(\boldsymbol{L}_{ij} \neq 0) \mathbb{E}_{q(\boldsymbol{\nu}_{j\boldsymbol{L}_{ij}})} \boldsymbol{\nu}_{j\boldsymbol{L}_{ij}}. \end{aligned} \tag{14}$$

Note that in $\boldsymbol{\nu}_{j\boldsymbol{L}_{ij}}$, the $\boldsymbol{L}_{ij}$ acts as the second dimension index of $\boldsymbol{\nu}$, i.e., when $\boldsymbol{L}_{ij} = k$, $\boldsymbol{\nu}_{j\boldsymbol{L}_{ij}}$ becomes $\boldsymbol{\nu}_{jk}$. By plugging $\boldsymbol{\eta}^*_{\boldsymbol{Y}}$ back into $\mathcal{L}$, we can define the final optimization objective as follows:

$$\mathcal{J}(\boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma}) \triangleq \mathcal{L}(\boldsymbol{\eta}^*_{\boldsymbol{Y}}, \boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma}). \tag{15}$$

It is proved by the SVAE [30] that $\mathcal{J}(\boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma})$ lower bounds the partially optimized mean field objective, i.e., $\max_{\boldsymbol{\eta}_{\boldsymbol{Y}}} \mathcal{L}(\boldsymbol{\eta}_{\boldsymbol{Y}}, \boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma}) \geqslant \mathcal{J}(\boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma})$; thus, $\mathcal{J}$ can serve as the variational lower bound of $\mathcal{L}$. As per [30], the natural gradient of $\mathcal{J}$ with respect to $\boldsymbol{\eta}_{\boldsymbol{\Theta}}$ is derived as follows:

$$\widetilde{\nabla}_{\boldsymbol{\eta}_{\boldsymbol{\Theta}}} \mathcal{J} = [\boldsymbol{\eta}^0_{\boldsymbol{\Theta}} + \mathbb{E}_{q^*(\boldsymbol{Y})}(\boldsymbol{t}(\boldsymbol{Y}, \boldsymbol{X}, \boldsymbol{L}), \boldsymbol{1}) - \boldsymbol{\eta}_{\boldsymbol{\Theta}}] + (\nabla_{\boldsymbol{\eta}(\boldsymbol{Y})} \mathcal{L}(\boldsymbol{\eta}^*_{\boldsymbol{Y}}, \boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma}), \boldsymbol{0}). \tag{16}$$

Here $\boldsymbol{\eta}^0_{\boldsymbol{\Theta}}$ is the prior natural parameter value of $\boldsymbol{\Theta}$ set by users. For our problem, the natural gradients for $\widetilde{\nabla}_{\boldsymbol{\eta}_{\boldsymbol{\pi}}} \mathcal{J}$ and $\widetilde{\nabla}_{\boldsymbol{\eta}_{\boldsymbol{\nu}_{jk}}} \mathcal{J}$ are, respectively, derived as follows:

$$\widetilde{\nabla}_{\boldsymbol{\eta}_{\boldsymbol{\pi}}} \mathcal{J} = \boldsymbol{\eta}^0_{\boldsymbol{\pi}} + \sum_{i=1}^{N} \mathbb{E}_{q^*(\boldsymbol{y}_i)} \boldsymbol{t}(\boldsymbol{y}_i) - \boldsymbol{\eta}_{\boldsymbol{\pi}}, \tag{17}$$

$$\widetilde{\nabla}_{\boldsymbol{\eta}_{\boldsymbol{\nu}_{jk}}} \mathcal{J} = \boldsymbol{\eta}^0_{\boldsymbol{\nu}_{jk}} + \sum_{i=1}^{N} \mathcal{I}(\boldsymbol{L}_{ij} \neq 0) \mathbb{E}_{q^*(\boldsymbol{y}_i)} \boldsymbol{t}(\boldsymbol{y}_i) \otimes \bar{\boldsymbol{L}}_{ij} - \boldsymbol{\eta}_{\boldsymbol{\nu}_{jk}}. \tag{18}$$

Here, $\bar{\boldsymbol{L}}_{ij}$ is the one-hot vector representation of $\boldsymbol{L}_{ij}$ and for parameters $\boldsymbol{\gamma}$, their gradient $\nabla_{\boldsymbol{\gamma}} \mathcal{J}$ can be directly computed in the DNN backpropagation framework. The complete optimization process of our BayesDGC is shown in Algorithm 1.

Algorithm 1 Bayesian deep generative crowdsourcing (BayesDGC)

Input: example features $\boldsymbol{X} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, crowd annotations $\boldsymbol{L} \in \{0, 1, \ldots, K\}^{N \times W}$, initial value for the global variational parameters $\boldsymbol{\eta}_{\boldsymbol{\Theta}}$ and neural network parameters $\boldsymbol{\gamma}$.

1: **Repeat**:
2: Given $\boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma}$, update the true label's natural parameter $\boldsymbol{\eta}^*_{\boldsymbol{y}_i}$ for each example using Eq. (14);
3: Given estimated $\boldsymbol{\eta}^*_{\boldsymbol{y}_i}$, compute the gradient of $\boldsymbol{\eta}_{\boldsymbol{\Theta}}$ using Eqs. (17) and (18) and $\boldsymbol{\gamma}$ via DNN back propagation, then conduct SGD updating for them;
4: **Until** the lower bound $\mathcal{J}(\boldsymbol{\eta}_{\boldsymbol{\Theta}}, \boldsymbol{\gamma})$ converges or the maximum number of epochs is reached.

Once the training finished, we estimated each example's true label assignment and each worker's confusion matrix using their respective expected sufficient statistics.

$$\mathbb{E}_{q(\boldsymbol{y})} \boldsymbol{t}(\boldsymbol{y}) = \begin{bmatrix} \boldsymbol{\pi}_1 \\ \vdots \\ \boldsymbol{\pi}_K \end{bmatrix}, \quad \mathbb{E}_{q(\boldsymbol{\nu}_{jk})} \boldsymbol{t}(\boldsymbol{\nu}_{jk}) = \begin{bmatrix} \varphi(\boldsymbol{\beta}_{jk1}) \\ \vdots \\ \varphi(\boldsymbol{\beta}_{jkK}) \end{bmatrix} - \varphi\left(\sum_{l=1}^{K} \boldsymbol{\beta}_{jkl} \right). \tag{19}$$

Here φ is the digamma function. For our classification problem, the DNN model with parameter $\boldsymbol{\gamma}$ can act as a learned classifier. For new data with features as input, their label probability can be predicted by applying the softmax function the output of the DNN model.

4 Experiments

In this section, we compare the proposed approach with several baseline methods on real-world crowdsourcing datasets.

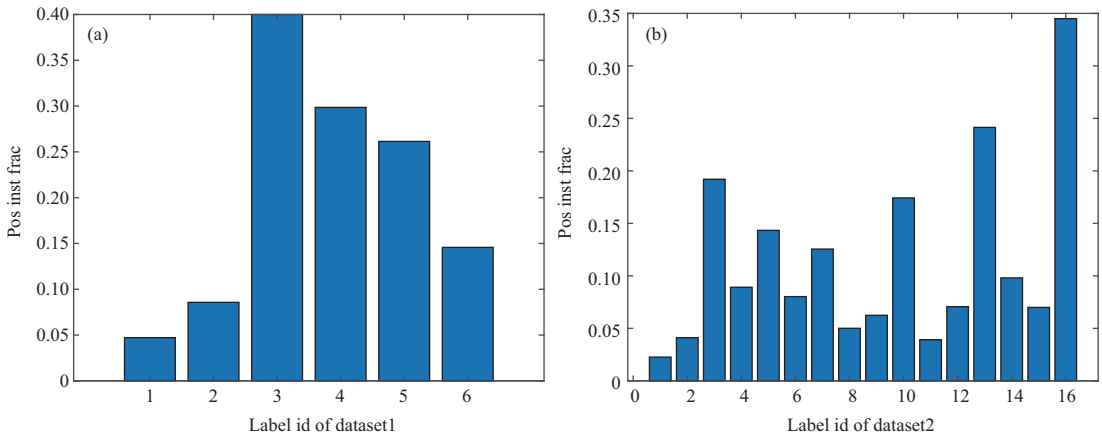

Figure 2 (Color online) The positive instance fraction of each label for (a) dataset1 and (b) dataset2.

Data sets. We use two image crowdsourcing datasets that we previously collected for the multi-label crowdsourcing study [32]. These are labeled dataset1 and dataset2, which, respectively, comprises 6, 16 candidate labels and 700, 1495 images, with ground truth labels annotated by human volunteers. The data analysis in [32] shows that the crowds' macro F1 scores vary primarily approximately $[0.70, 0.80]$, which indicates the reliability of the majority of workers and establishes the basis of learning feasibility.

In this study, we conducted experiments on each label independently; accordingly, we derived 22 binary datasets. Originally, in [32], the annotations of 18 and 15 workers were maintained for experiments. Observation of the results in [32] and our experiments show that the performance of most methods converges when the number of workers exceeds 10. In this study, for experiment efficiency, we maintained the annotations of 9 workers who annotated the most data and conducted aggregation. The original 1248-dim Fisher vector features were used.

Comparison methods. We compared four representative state-of-the-art crowdsourcing methods, i.e., MV, DS [7], MaxEn [9], and Yutc [4]. Furthermore, for our proposed BayesDGC model, we implement the non-deep Bayesian variant, BayesGC, for which the DNN classifier prior is not used.

For the proposed BayesDGC and BayesGC, the Dirichlet prior $\mathrm{Dir}(\boldsymbol{\pi}|\boldsymbol{\alpha}^0)$ with $\boldsymbol{\alpha}^0 = 1.1$ is used for $\boldsymbol{\pi}$. For $\boldsymbol{\nu}_{jk}$, one similar prior is used for all workers and $\mathrm{Dir}(\boldsymbol{\nu}_{jk}|\boldsymbol{\beta}^0)$ with $\boldsymbol{\beta}^0_{kk} = 5$, $\boldsymbol{\beta}^0_{kk'} = 2$, $k \neq k'$ is used to encode that workers are better compared with random guessing. For BayesDGC, a single hidden layer (with 100 nodes) multilayer perceptron is exploited as the deep neural network classifier. The Adam optimizer with 0.001 learning rate is used. For the baselines, we use codes provided by their authors and the default parameter suggested for each is used. Except for DS, the two-coin model implemented by [19] is used.

To test the performance of the approaches' dependence on the number of annotations, we vary the observed fraction of annotations p from 10% to 100% in a uniformly random manner and report the average and standard deviation results for 10 times repetitions. As the datasets were originally for multi-label tasks, the 22 binary data were significantly imbalanced, the positive instance fractions of which are shown in Figure 2. Most labels were extremely imbalanced; thus, to conduct evaluation of the results, we treated the top k ranked labels of each method as its positive prediction and the rest as negative predictions. Here, k is the number of true positive labels for each label. The accuracy and F1 score results are reported. In future, we aim to design algorithms that consider the imbalance factor for crowdsourcing learning.

4.1 Results

For the 22 datasets, we use l to denote the corresponding binary classification task for the l-th label of dataset1 and dataset2. Figure 3 shows the accuracy results. It is shown that the proposed BayesDGC significantly outperforms other methods in most cases, whereas BayesGC is inferior. For all approaches, an overall monotonically increasing performance is observed as the number of annotations increases. Note that the four annotation-only exploitation methods, i.e., MV, DS, MaxEn, and BayesGC, often achieve very similar performances and exhibit an obvious gap compared to when using BayesDGC, particularly when the number of annotations is limited. This indicates that as an essential information source,

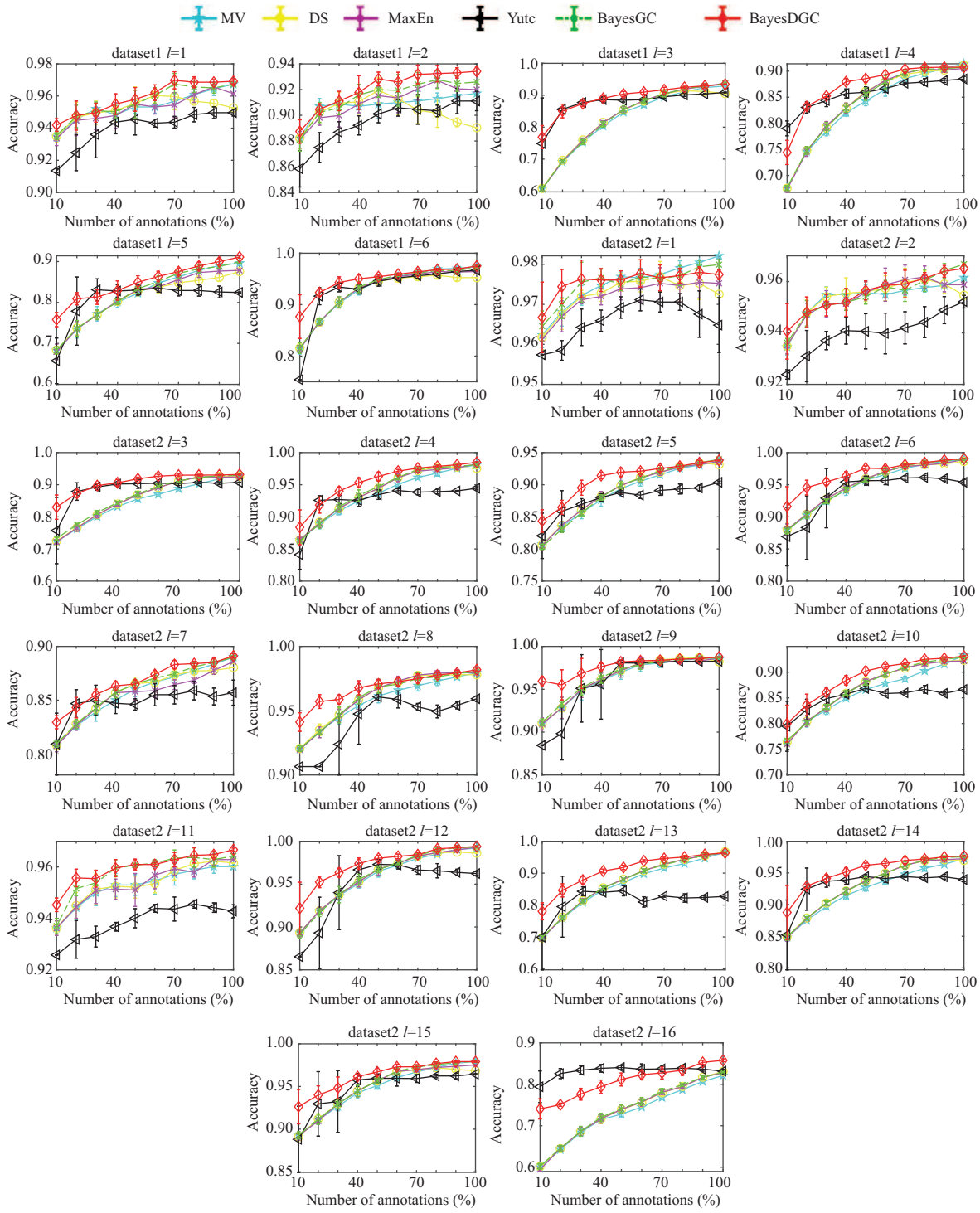

Figure 3 (Color online) Accuracy results of all methods on 22 real-word datasets.

data features should not be ignored. We then considered the performance of Yutc, which exploited the feature information through a logistic regression classifier. On some datasets such as dataset1 $l = 3, 6$ and dataset2 $l = 3, 6, 16$, Yutc achieves comparable results or even better. However, these results are not stable and in certain cases even show the worst performance, e.g., dataset1 $l = 1, 2$ and dataset2 $l = 1, 2, 8, 11$, which may have been attributed to linear model inefficiency or improper parameter setting. This indicates that for specific applications, careful parameter tuning must be effected. However, for our fully Bayesian deep model, the DNN learning feature provides sufficient model capacity and the parameter tuning is automatically conducted.

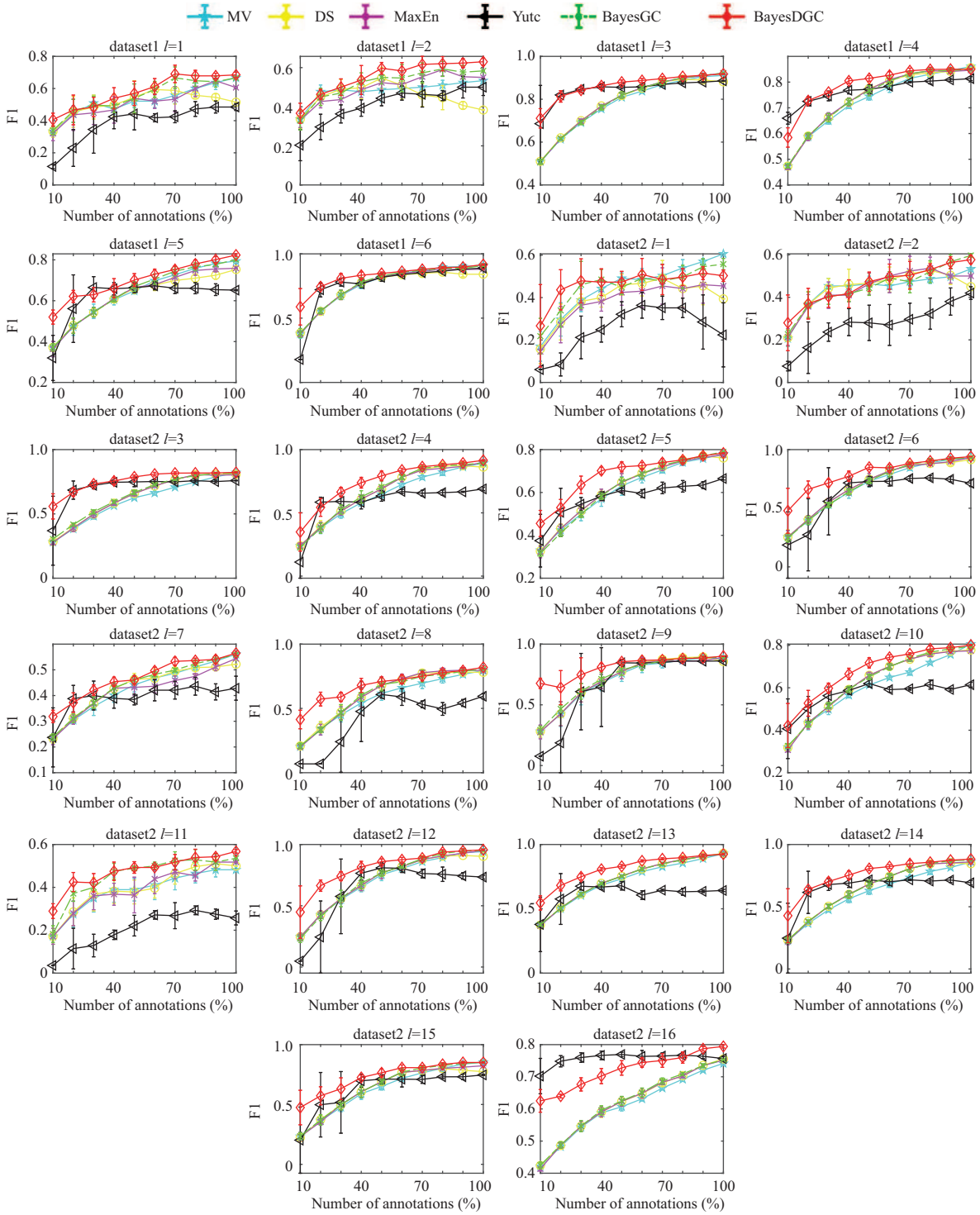

Figure 4 (Color online) F1 score results of all methods on 22 real-word datasets.

Figure 4 shows the results for F1 score. The comparison is similar to the accuracy result, but with much lower performance, indicating the class imbalance challenge for crowdsourcing learning.

5 Conclusion

In this study, we propose a BayesDGC model for classification problem. The model comprises a probabilistic annotation generation process, and a deep neural network model for effective representational

learning. To address the inference challenge, we implement an efficient end-to-end natural gradient stochastic variational inference algorithm, that avoids the computational overhead of EM and sampling approaches and concurrently retains the interpretable probabilistic structure. Experiments indicated the superiority of the proposed approach. In future, we aim to extend the sufficiently general inference algorithm to more sophisticated crowdsourcing aggregation problems such as annotation correlation modeling and extension to multi-label tasks.

Acknowledgements This work was supported by Fundamental Research Funds for the Central Universities (Grant No. NJ20190-10), National Natural Science Foundation of China (Grant No. 61906089), Jiangsu Province Basic Research Program (Grant No. BK20190408), and China Postdoc Science Foundation (the First Pre-station Special Grant).

References

1 Horvitz E. Reflections on challenges and promises of mixed-initiative interaction. AI Mag, 2007, 28: 13–22
2 Weld D, Lin C, Bragg J. Artificial intelligence and collective intelligence. In: Proceedings of Collective Intelligence Handbook. 2015
3 Snow R, O'Connor B, Jurafsky D, et al. Cheap and fast — but is it good? Evaluating non-expert annotations for natural language tasks. In: Proceedings of Conference on Empirical Methods in Natural Language Processing, 2008. 254–263
4 Raykar V, Yu S, Zhao L, et al. Learning from crowds. J Mach Learn Res, 2010, 11: 1297–1322
5 Welinder P, Branson S, Belongie S, et al. The multidimensional wisdom of crowds. In: Proceedings of Advances in Neural Information Processing Systems, 2010. 2024–2432
6 Li Q Q, Li Y L, Gao J, et al. A confidence-aware approach for truth discovery on long-tail data. Proc VLDB Endow, 2014, 8: 425–436
7 Dawid A P, Skene A M. Maximum likelihood estimation of observer error-rates using the EM algorithm. Appl Stat, 1979, 28: 20
8 Whitehill J, Ruvolo P, Wu T, et al. Whose vote should count more: optimal integration of labels from labelers of unknown expertise. In: Proceedings of Advances in Neural Information Processing Systems, 2009. 2035–2043
9 Zhou D Y, Basu S, Mao Y, et al. Learning from the wisdom of crowds by minimax entropy. In: Proceedings of Advances in Neural Information Processing Systems, 2012. 2195–2203
10 Li Y, Rubinstein B I P, Cohn T. Exploiting worker correlation for label aggregation in crowdsourcing. In: Proceedings of the 36th International Conference on Machine Learning, 2019. 3886–3895
11 Rodrigues F, Pereira F, Ribeiro B, et al. Gaussian process classification and active learning with multiple annotators. In: Proceedings of the 31st International Conference on Machine Learning, 2014. 433–441
12 Albarqouni S, Baur C, Achilles F, et al. AggNet: deep learning from crowds for mitosis detection in breast cancer histology images. IEEE Trans Med Imag, 2016, 35: 1313–1321
13 Atarashi K, Oyama S, Kurihara M. Semi-supervised learning from crowds using deep generative models. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018. 1555–1562
14 Rodrigues F, Pereira P C. Deep learning from crowds. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018. 1611–1618
15 Tanno R, Saeedi A, Sankaranarayanan S, et al. Learning from noisy labels by regularized estimation of annotator confusion. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2019. 11244–11253
16 Imamura H, Sato I, Sugiyama M. Analysis of minimax error rate for crowdsourcing and its application to worker clustering model. In: Proceedings of the 35th International Conference on Machine Learning, 2018. 2152–2161
17 Sheng V S, Zhang J, Gu B, et al. Majority voting and pairing with multiple noisy labeling. IEEE Trans Knowl Data Eng, 2019, 31: 1355–1368
18 Tao F N, Jiang L X, Li C Q. Label similarity-based weighted soft majority voting and pairing for crowdsourcing. Knowl Inf Syst, 2020, 62: 2521–2538
19 Liu Q, Peng J, Ihler A. Variational inference for crowdsourcing. In: Proceedings of Advances in Neural Information Processing Systems, 2012. 692–700
20 Kim H C, Ghahramani Z. Bayesian classifier combination. In: Proceedings of the 15th International Conference on Artificial Intelligence and Statistics, 2012. 619–627
21 Simpson E, Roberts S, Psorakis I, et al. Dynamic Bayesian combination of multiple imperfect classifiers. In: Decision Making and Imperfection. Berlin: Springer, 2013
22 Venanzi M, Guiver J, Kazai P, et al. Community-based bayesian aggregation models for crowdsourcing. In: Proceedings of the 23rd International Conference on World Wide Web, 2014. 155–164
23 Moreno P G, Artes-Rodrguez A, Teh Y W, et al. Bayesian nonparametric crowdsourcing. J Mach Learn Res, 2015, 16: 1607–1627
24 Rodrigues F, Lourenco M, Ribeiro B, et al. Learning supervised topic models for classification and regression from crowds. IEEE Trans Pattern Anal Mach Intell, 2017, 39: 2409–2422
25 Zhang H, Jiang L X, Xu W Q. Multiple noisy label distribution propagation for crowdsourcing. In: Proceedings of the 28th

International Joint Conference on Artificial Intelligence, 2019. 1473–1479
26 LeCun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015, 521: 436–444
27 Luo Y C, Tian T, Shi J X, et al. Semi-crowdsourced clustering with deep generative models. In: Proceedings of Advances in Neural Information Processing Systems, 2018. 3216–3226
28 Kingma D P, Mohamed S, Rezende D J, et al. Semi-supervised learning with deep generative models. In: Proceedings of Advances in Neural Information Processing Systems, 2014. 3581–3589
29 Kingma D P, Welling M. Auto-encoding variational bayes. In: Proceedings of the 2nd International Conference on Learning Representations, 2014
30 Johnson M J, Duvenaud D, Wiltschko A B, et al. Composing graphical models with neural networks for structured representations and fast inference. In: Proceedings of Advances in Neural Information Processing Systems, 2016. 2946–2954
31 Hoffman M D, Blei D M, Wang C, et al. Stochastic variational inference. J Mach Learn Res, 2013, 14: 1303–1347
32 Li S Y, Jiang Y, Chawla N V, et al. Multi-label learning from crowds. IEEE Trans Knowl Data Eng, 2019, 31: 1369–1382