Variational Label Enhancement

Ning Xu [1] **Jun Shu** [2] **Yun-Peng Liu** [1] **Xin Geng** [1]

Abstract

Label distribution covers a certain number of labels, representing the degree to which each label describes the instance. When dealing with label ambiguity, label distribution could describe the supervised information in a fine-grained way. Unfortunately, many training sets only contain simple logical labels rather than label distributions due to the difficulty of obtaining label distributions directly. To solve this problem, we consider the label distributions as the latent vectors and infer them from the logical labels in the training datasets by using variational inference. After that, we induce a predictive model to train the label distribution data by employing the multi-output regression technique. The recovery experiment on fourteen label distribution datasets and the predictive experiment on ten multi-label learning datasets validate the advantage of our approach over the state-of-the-art approaches.

1. Introduction

Learning with ambiguity is a hot topic in recent machine learning and data mining research. A learning process is essentially building a mapping from the instances to the labels. This paper mainly focuses on the ambiguity at the label side of the mapping, i.e., one instance is not necessarily mapped to one label (Tsoumakas & Katakis, 2006). During the past decade, the techniques for learning with label ambiguity have been widely employed to learn from data with rich semantics, such as text (Rubin et al., 2012), image (Cabral et al., 2011), audio (Lo et al., 2011), video (Wang et al., 2011), etc. For learning with label ambiguity, the logical labels are always assigned to the instance, partitioning the supervised information into relevance/irrelevance labels rigidly (Gibaja & Ventura, 2015; Zhang & Zhou, 2014).

[1]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China [2]School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China. Correspondence to: Xin Geng <xgeng@seu.edu.cn>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

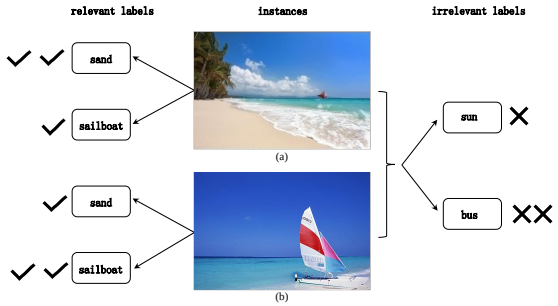

Figure 1. An example of the relative importance among relevant / irrelevant labels

However, the relevance or irrelevance of a label to an instance is essentially relative in the real-world tasks. When multiple labels are associated with an instance, the relative importance among them is more likely to be different rather than exactly equal. For instance, when "sand" and "sailboat" are relevant to the two images in Fig. 1, "sand" is more significant than "sailboat" in image (a) and the opposite scenario occurs in image (b). On the other hand, the "irrelevance" of each irrelevant label may be very different. For instance, "bus" is more irrelevant than "sun" to the two images in Fig. 1, since "sun" often appears with "sand"and "sailboat" on the beach. Therefore, assigning the logical label $l_{\boldsymbol{x}}^{y} \in \{0, 1\}$ to each instance $\boldsymbol{x}$ with the relevant (irrelevant) label y ignores the relative importance among the relevant (irrelevant) labels.

To solve this problem, a more natural way to label an instance $\boldsymbol{x}$ is to assign a real number $d_{\boldsymbol{x}}^{y}$ to each possible label y, representing the degree to which y describes $\boldsymbol{x}$. Such $d_{\boldsymbol{x}}^{y}$ is called the *description degree* of y to $\boldsymbol{x}$. For a particular instance, the description degrees of all the labels constitute a real-valued vector called *label distribution* (Geng, 2016). Therefore, label distribution is more fine-grained to describe the supervised information in the tasks of learning with label ambiguity.

However, label distributions are not explicitly available in most training sets as quantifying the description degrees is costly. It needs to be somehow recovered from the training set, a process which is named as label enhancement (LE) (Xu et al., 2018). After the label distributions are recov-

ered, more effective supervised learning can be achieved by leveraging the label distributions (Li et al., 2015; Hou et al., 2016). Note that although some label enhancement methods have been proposed (Xu et al., 2019; Li et al., 2015; Hou et al., 2016), there is no theoretical explanation about the recovered label distribution and the process of label enhancement.

In this paper, a theoretical explanation about the essence of label enhancement is proposed. By inducing the generative model of the label distribution and adopt the variational inference technique, a variational lower bound of the label distribution is given and a novel LE approach called Label Enhancement via Variational Inference (LEVI) is proposed to infer the label distributions from the logical labels. In addition, we show how our method can support multi-label learning (Tsoumakas & Katakis, 2006), and evaluate it against several state-of-the-art methods.

The rest of this paper is organized as follows. Firstly, some related work is briefly reviewed and discussed in Section 2. Secondly, technical details of the theoretical explanation and proposed approach LEVI are introduced in Section 3. Then, LEVI for multi-label learning is proposed in Section4. After that, the results of the comparative experiments are reported in Section 5. Finally, conclusions are drawn in Section 6.

2. Related Work

Label distribution explicitly models label ambiguity with the description degree, which is not the probability that y correctly labels $\boldsymbol{x}$, but the proportion that y accounts for in a full class description of $\boldsymbol{x}$. Therefore, label distribution can be distinguished from the previous studies on probabilistic labels (Quost & Denœux, 2009; Denœux & Zouhal, 2001; Smyth et al., 1995), where the basic assumption is that only one 'correct' label is assigned to each instance. Probabilistic labels are mainly used when the real label of the instance cannot be obtained with certainty. In practice, it is usually difficult to determine the probability (or confidence) of a label. In most cases, it relies on the prior knowledge of the human experts, which is a highly subjective and variable process. As a result, the problem of learning from probabilistic labels has not been extensively studied to date.

From the conceptual point of view, it is worthwhile to distinguish description degree from the concept membership used in fuzzy classification. Membership is designed to handle the status of partial truth, which is a truth value which ranges between completely true and completely false. On the other hand, description degree reflects the ambiguity of the label description of the instance, i.e., one label may only partially describe the instance, but it is completely true that the label describes the instance. Fortunately, although the concept of membership is fundamentally different from description degree, some methods (Gayar et al., 2006; Jiang et al., 2006) which focus on generating membership can be applied to generate label distributions(Xu et al., 2019).

Label distribution learning (LDL) is a novel learning paradigm, which labels an instance with a label distribution and learns a mapping from instance to label distribution straightly. LDL has been successfully applied to many real applications, such as facial landmark detection (Su & Geng, 2019), age estimation (Gao et al., 2018; Geng et al., 2013), head pose estimation (Geng & Xia, 2014), multi-label ranking for natural scene images (Geng & Luo, 2014), zero-shot Learning (Huo & Geng, 2017) and emotion analysis from texts (Zhou et al., 2016). According to the theoretical analysis (Wang & Geng, 2019), LDL is approximate to the optimal classifier via learning on the instances labeled by the ground-truth label distributions. However, in most training sets, the label distribution is not explicitly available. There are few work to deal with this situation. One recent paper (Li et al., 2015) adopts the propagation technique to generate the label distributions without considering the correlations between the labels.

Label enhancement (LE) is a process to recover the label distributions from the logical labels in the training datasets. GLLE (Xu et al., 2018), LP (Li et al., 2015) and ML (Hou et al., 2016) are three representative algorithms of LE. They assume that the label distribution space should share similar local topological structure in the feature space. GLLE constructs a local similarity matrix to preserve the topological structure information of the feature space, LP adopts label propagation technique to propagate labeling-importance information, ML adopts the local linear embedding technique to achieve identified label degrees. Nonetheless, these methods all rely on the smoothness assumption (Zhu et al., 2005), i.e., the points close to each other are more likely to share a label. This assumption, however, might restrict modeling capacity, as graph edges need to be necessarily encoded which introduces additional bias.

In the next section, a novel label enhancement approach will be introduced. Different from existing label enhancement approaches, the generative model of the label distribution is proposed and the label distribution could be recovered via variational inference with limited assumption. The predictive model by employing multi-output regression techniques is also induced to leveraging the recovered label distributions for multi-label learning.

3. The LEVI Method

First of all, the main notations used in this paper are listed as follows. The instance variable is denoted by $\boldsymbol{x}$, the particular i-th instance is denoted by $\boldsymbol{x}_i$, the label variable is denoted

by y, the particular j-th label value is denoted by y_j, the logical label vector of $\boldsymbol{x}_i$ is denoted by $\boldsymbol{l}_i = (l_{\boldsymbol{x}_i}^{y_1}, l_{\boldsymbol{x}_i}^{y_2}, ..., l_{\boldsymbol{x}_i}^{y_c})^\top$, where c is the number of possible labels. The description degree of y to $\boldsymbol{x}$ is denoted by $d_{\boldsymbol{x}}^{y}$, and the label distribution of $\boldsymbol{x}_i$ is denoted by $\boldsymbol{d}_i = (d_{\boldsymbol{x}_i}^{y_1}, d_{\boldsymbol{x}_i}^{y_2}, ..., d_{\boldsymbol{x}_i}^{y_c})^\top$. Let $\mathcal{X} = \mathbb{R}^q$ denote the q-dimensional feature space.

3.1. Variational Lower Bound

Since the difficulty and costly of quantifying the label distributions, people instead, choose simplifying the supervised information by the logical labels. Therefore, the logical labels are observed discrete vectors and the label distribution are latent vectors. We assume that the label distribution is generated from some prior distribution $p(\boldsymbol{d})$, and then the logical label vector $\boldsymbol{l}$ is generated from some conditional distribution $p(\boldsymbol{l}|\boldsymbol{d})$.

Computation of the exact posterior distribution is intractable due to the nonlinear, non-conjugate dependencies between the random variables. To allow for tractable and scalable inference and parameter learning, variational inference is adopted. We introduce a fixed-form distribution $q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$ with parameters $\boldsymbol{w}$ that approximates the true posterior distribution $p(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$. We then follow the variational principle to derive a lower bound on the marginal likelihood of the model. This bound forms our objective function and ensures that our approximate posterior is as close as possible to the true posterior.

We begin with the definition of Kullback-Leibler divergence (KL divergence) between $p(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$ and $q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$:

$$\begin{aligned}\mathrm{KL}[q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})||p(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})] = \mathbb{E}_{q(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})}[\log q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x}) \\ - \log p(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})].\end{aligned} \tag{1}$$

Applying Bayes rule:

$$\begin{aligned}\mathrm{KL}[q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})||p(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})] = \mathbb{E}_{q(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})}[\log q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x}) - \\ \log p(\boldsymbol{l}|\boldsymbol{d}) - \log p(\boldsymbol{x}|\boldsymbol{d}) - \log p(\boldsymbol{d}) + \log p(\boldsymbol{l}, \boldsymbol{x})].\end{aligned} \tag{2}$$

Here, $\log p(\boldsymbol{l}, \boldsymbol{x})$ comes out of the expectation because it dose not depend on $\boldsymbol{d}$:

$$\begin{aligned}\mathrm{KL}[q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})||p(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})] = \log p(\boldsymbol{l}, \boldsymbol{x}) - \mathbb{E}_{q(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})} \\ [\log p(\boldsymbol{l}|\boldsymbol{d}) + \log p(\boldsymbol{x}|\boldsymbol{d})] + \mathrm{KL}[q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})||p(\boldsymbol{d})].\end{aligned} \tag{3}$$

Since this KL-divergence is non-negative, we have :

$$\begin{aligned}\log p(\boldsymbol{l}, \boldsymbol{x}) \geq \mathbb{E}_{q(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})}[\log p(\boldsymbol{l}|\boldsymbol{d}) + \log p(\boldsymbol{x}|\boldsymbol{d})] \\ -\mathrm{KL}[q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})||p(\boldsymbol{d})].\end{aligned} \tag{4}$$

We construct the approximate posterior distribution q as an inference model, which has become a popular approach for efficient variational inference (Kingma & Welling, 2014; Rezende et al., 2014). Using an inference network, we avoid the need to compute per data point variational parameters, but can compute a set of global variational parameters instead. This allows us to amortise the cost of inference by generalizing between the posterior estimates for all latent variables through the parameters of the inference model, and allows for fast inference at both training and testing time. Then, the ELBO (Evidence Lower Bound) is written as

$$\begin{aligned}\mathcal{L}(\boldsymbol{x}, \boldsymbol{l}; \boldsymbol{\vartheta}, \boldsymbol{\eta}, \boldsymbol{w}) = \mathbb{E}_{q_{\boldsymbol{w}}(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})}[\log p_{\boldsymbol{\vartheta}}(\boldsymbol{l}|\boldsymbol{d}) + \log p_{\boldsymbol{\eta}}(\boldsymbol{x}|\boldsymbol{d})] \\ -\mathrm{KL}[q_{\boldsymbol{w}}(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})||p(\boldsymbol{d})].\end{aligned} \tag{5}$$

Here the inference network is introduced for $q(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$, and we parameterize them as deep neural networks whose outputs form the parameters of the distribution $q_{\boldsymbol{w}}(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$. The logical label vector $\boldsymbol{l}$ and the instance $\boldsymbol{x}$ are generated from the deep neural networks distribution $p_{\boldsymbol{\vartheta}}(\boldsymbol{l}|\boldsymbol{d})$ and $p_{\boldsymbol{\eta}}(\boldsymbol{x}|\boldsymbol{d})$, respectively.

3.2. Label Enhancement Objective

The bound in Eq. (5) provides a unified objective function for optimisation of all the parameters $\boldsymbol{w}$, $\boldsymbol{\vartheta}$ and $\boldsymbol{\phi}$ of the generative and inference models. By expanding the label distribution to $\boldsymbol{d} \in R^c$, we assume that the prior over the latent label distribution be the centered isotropic multivariate Gaussian $\mathcal{N}(\mathbf{0}, \mathbf{I})$. We let the variational approximate posterior be a multivariate Gaussian with a diagonal covariance structure $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where the mean and covariance matrix of the approximate posterior, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, are outputs of the MLP with parameter $\boldsymbol{w}$. Then the KL divergence in the ELBO can be computed:

$$\begin{aligned}\mathrm{KL}[q_{\boldsymbol{w}}(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})||p(\boldsymbol{d})] = \frac{1}{2}\{\mathrm{tr}(\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\top\boldsymbol{\mu} - k \\ - \log|\boldsymbol{\Sigma}|\}.\end{aligned} \tag{6}$$

where k is the dimensionality of the distribution.

Then, we assume $p_{\boldsymbol{\vartheta}}(\boldsymbol{l}|\boldsymbol{d})$ be a multivariate Bernoulli whose probabilities $\boldsymbol{\tau}$ are computed from $\boldsymbol{d}$ with the MLP parameterized by $\boldsymbol{\vartheta}$, and $p_{\boldsymbol{\eta}}(\boldsymbol{x}|\boldsymbol{d})$ be a multivariate Gaussian whose means $\boldsymbol{\rho}$ are computed from $\boldsymbol{d}$ with the MLP parameterized by $\boldsymbol{\eta}$. Then the first part of the ELBO can be computed:

$$\begin{aligned}\mathbb{E}_{q_{\boldsymbol{w}}(\boldsymbol{d}|\boldsymbol{l},\boldsymbol{x})}[\log p_{\boldsymbol{\vartheta}}(\boldsymbol{l}|\boldsymbol{d}) + \log p_{\boldsymbol{\eta}}(\boldsymbol{x}|\boldsymbol{d})] = \\ \frac{1}{L}\sum_{m=1}^{L}\sum_{i=1}^{c} l_i \log \tau_i^{(m)} + (1 - l_i) \cdot \log\left(1 - \tau_i^{(m)}\right) \\ -\frac{1}{L}\sum_{m=1}^{L}\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{\rho}^{(m)}\|_2^2.\end{aligned} \tag{7}$$

Note that back-propagate the error through a layer that samples $\boldsymbol{d}$ from $q_{\boldsymbol{w}}(\boldsymbol{d}|\boldsymbol{l}, \boldsymbol{x})$, which is a non-continuous operation and has no gradient. In order to to move the sampling to

an input layer ,the reparameterization trick (Rezende et al., 2014) is induced to sample $\boldsymbol{d}$ by:

$$\boldsymbol{d} = \boldsymbol{\mu} + \boldsymbol{\Sigma}\boldsymbol{\epsilon}, \tag{8}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. In this case, Eq. (7) can be computed and differentiated.

Since the label distributions inherit from the initial labels relevance and irrelevance, we add the least squares for the label distribution and the initial labels into the objective function. Then, we formulate the label enhancement problem into an optimization framework over Eq. (7) and Eq. (6), and yields the target function for minimization:

$$\begin{aligned} T(\boldsymbol{\vartheta}, \boldsymbol{\eta}, \boldsymbol{w}) =& \frac{1}{L}\sum_{m=1}^{L}\frac{1}{2}\|\boldsymbol{x} - \boldsymbol{\rho}^{(m)}\|_2^2 + \lambda\|\boldsymbol{d}^{(m)} - \boldsymbol{l}\|_2^2 \\ &- \sum_{i=1}^{c} l_i \log \tau_i^{(m)} + (1 - l_i) \cdot \log\left(1 - \tau_i^{(m)}\right) \\ &+ \frac{1}{2}\{\mathrm{tr}(\boldsymbol{\Sigma}) + \boldsymbol{\mu}^\top\boldsymbol{\mu} - k - \log|\boldsymbol{\Sigma}|\}, \end{aligned} \tag{9}$$

where λ is a hyper-parameter, $\boldsymbol{\Sigma} = \mathrm{MLP}_{\boldsymbol{\Sigma}}(\boldsymbol{l}, \boldsymbol{x}; \boldsymbol{w})$, $\boldsymbol{\mu} = \mathrm{MLP}_{\boldsymbol{\mu}}(\boldsymbol{l}, \boldsymbol{x}; \boldsymbol{w})$, $\boldsymbol{\tau}^{(m)} = \mathrm{MLP}_{\boldsymbol{\tau}}(\boldsymbol{d}^{(m)}; \boldsymbol{\vartheta})$, $\boldsymbol{\rho}^{(m)} = \mathrm{MLP}_{\boldsymbol{\rho}}(\boldsymbol{d}^{(m)}; \boldsymbol{\eta})$, $\boldsymbol{d}^{(m)} = \boldsymbol{\mu} + \boldsymbol{\Sigma}\boldsymbol{\epsilon}^{(m)}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.

During optimization we use the standard stochastic gradient based optimization methods such as SGD or AdaGrad. After $\boldsymbol{\vartheta}, \boldsymbol{\eta}, \boldsymbol{w}$ are determined, the label distribution $\boldsymbol{d}_i$ of each instance $\boldsymbol{x}_i$ is sampled from the posterior $\boldsymbol{d}_i \sim q_{\boldsymbol{w}}(\boldsymbol{d}|\boldsymbol{l}_i, \boldsymbol{x}_i)$. Finally, we normalize $\boldsymbol{d}_i$ by using the softmax normalization.

4. LEVI for Multi-Label Learning

In this section, LEVI is leveraged for multi-label learning (MLL) (Zhang & Zhou, 2014), which is the most representative learning paradigm for learning with label ambiguity. When the label distribution $\boldsymbol{d}_i$ of each $\boldsymbol{x}_i$ has been recovered by LEVI, the original MLL training set can be transformed into $\mathcal{E} = \{(\boldsymbol{x}_i, \boldsymbol{d}_i)|1 \le i \le n\}$. Then, we generalize a regressor to solve the multi-dimensional case. In addition, our regressor not only concerns the distance between the predicted and the real values, but also the sign consistency of them. It leads to the minimization of

$$\Omega(\boldsymbol{\Theta}, \boldsymbol{b}) = \frac{1}{2}\sum_{j=1}^{c}\|\boldsymbol{\theta}^j\|^2 + C_1\sum_{i=1}^{n}\Omega_{1i} + C_2\sum_{i=1}^{n}\Omega_{2i}, \tag{10}$$

where $\boldsymbol{\Theta} = [\boldsymbol{\theta}^1, ..., \boldsymbol{\theta}^c]$, $\boldsymbol{b} = [b^1, ..., b^c]$, Ω_1 and Ω_2 are the regression loss and the sign loss, respectively.

As shown in Eq. (10), the first term of $\Omega(\boldsymbol{\Theta}, \boldsymbol{b})$ controls the complexity of the induced model. In addition, the second term of $\Omega(\boldsymbol{\Theta}, \boldsymbol{b})$ is defined to consider all dimensions into a unique restriction and yield a single support vector for all dimensions:

$$\Omega_{1i} = \begin{cases} 0 & r_i < \varepsilon \\ r_i^2 - 2r_i\varepsilon + \varepsilon^2 & r_i \ge \varepsilon, \end{cases} \tag{11}$$

where $r_i = \|\boldsymbol{e}_i\| = \sqrt{\boldsymbol{e}_i^\top \boldsymbol{e}_i}$, $\boldsymbol{e}_i = \boldsymbol{d}_i - \varphi(\boldsymbol{x}_i)^\top\boldsymbol{\Theta} - \boldsymbol{b}$. This will create an insensitive zone determined by ε around the estimate, i.e., the loss of r less than ε will be ignored. The third term is used to make the signs of the predictive output and the logical label same as much as possible:

$$\Omega_{2i} = -\sum_{j=1}^{c} l_i^j(\varphi(\boldsymbol{x}_i)^\top\boldsymbol{\theta}^j + b^j). \tag{12}$$

The meaning of Eq. (12) is that if the signs of the predictive output and the logical label are different, there will be some positive loss, otherwise the loss will be negative.

It is a piecewise quadratic problem whose optimum can be integrated as solving a system of linear equations for $j = 1, \ldots, c$:

$$\begin{bmatrix} C_1\boldsymbol{\Phi}^\top\boldsymbol{F}\boldsymbol{\Phi} + \boldsymbol{I} & C_1\boldsymbol{\Phi}^\top\boldsymbol{a} \\ C_1\boldsymbol{a}^\top\boldsymbol{\Phi} & C_1\mathbf{1}^\top\boldsymbol{a} \end{bmatrix}\begin{bmatrix} \boldsymbol{\theta}^j \\ b^j \end{bmatrix} = \begin{bmatrix} C_1\boldsymbol{\Phi}^\top\boldsymbol{F}\boldsymbol{d}^j + C_2\boldsymbol{\Phi}^\top\boldsymbol{l}^j \\ C_1\boldsymbol{a}^\top\boldsymbol{d}^j + C_2\mathbf{1}^\top\boldsymbol{l}^j \end{bmatrix}, \tag{13}$$

where $\boldsymbol{\Phi} = [\varphi(\boldsymbol{x}_1), ..., \varphi(\boldsymbol{x}_n)]^\top$, $\boldsymbol{a} = [a_1, ..., a_n]^\top$, $\boldsymbol{F}_i^k = a_i\delta_i^k$ (δ_i^k is the Kronecker's delta function), and $\boldsymbol{l}^j = [l_1^j, \ldots, l_n^j]^\top$. Then, the direction of the optimal solution of Eq. (13) is used as the descending direction for the optimization of $\Omega(\boldsymbol{\Theta}, \boldsymbol{b})$, and the solution for the next iteration ($\boldsymbol{\Theta}^{(k+1)}$ and $\boldsymbol{b}^{(k+1)}$) is obtained via a line search algorithm along this direction.

Finally, the predicted label set for unseen instance is determined via virtual label bipartition (Li et al., 2015). An extra virtual label y_0 is added into the original label set, i.e., the extended original label set $\mathcal{Y}' = \mathcal{Y} \cup \{y_0\} = \{y_0, y_1, ..., y_c\}$. In this paper, the origin value $l_{\boldsymbol{x}}^{y_0}$ is set to 0.5. Once the recovered label distribution and the predictive model have been learned on the extended original label set, the extended label distribution $\boldsymbol{d}^*$ corresponding to the test instance $\boldsymbol{x}^*$ can be predicted. Then, the predicted label set for $\boldsymbol{x}^*$ is determined as:

$$f(\boldsymbol{x}) = \{y_j \mid d_{\boldsymbol{x}}^{y_j} > d_{\boldsymbol{x}}^{y_0}, 1 \le j \le c\}. \tag{14}$$

5. Experiments

5.1. Recovery Experiment

We consider the following learning setting. With each instance, a label distribution is associated. The training set, however, contains for each instance not the actual distribution, but logical labels. As shown in Fig. 2, we recover

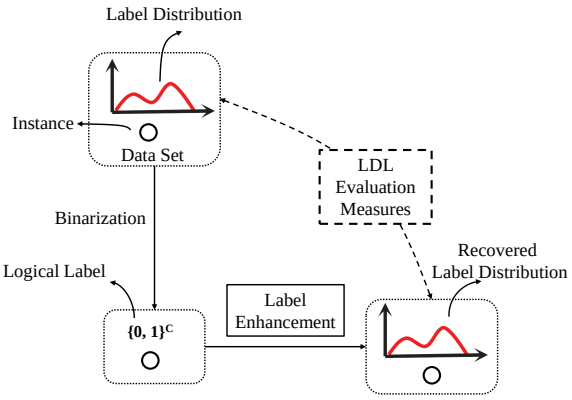

Figure 2. The schematic diagram of the recovery experiment.

the label distributions from the logical labels via the LE algorithms, and then compare the recovered label distributions with the ground-truth label distributions. The label set includes the labels with the highest weights in the distribution, and is the smallest set such that the sum of these weights exceeds a given threshold. This setting can model, for instance, the way in which users label images or add keywords to texts: it assumes that users add labels starting with the most relevant ones, until they feel the labeling is sufficiently complete.

The logical labels in the datasets can be binarized from the real label distributions as follows. For each instance $\boldsymbol{x}$, the greatest description degree $d_{\boldsymbol{x}}^{y_j}$ is found, and the label y_j is set to relevant label, i.e., $l_{\boldsymbol{x}}^{y_j} = 1$. Then, we calculate the sum of the description degrees of all the current relevant labels $H = \sum_{y_j \in \mathcal{Y}^+} d_{\boldsymbol{x}}^{y_j}$, where $\mathcal{Y}^+$ is the set of the current relevant labels. If H is less than a predefined threshold T, we continue finding the greatest description degree among other labels excluded from $\mathcal{Y}^+$ and select the label corresponding to the greatest description degree into $\mathcal{Y}^+$. This process continues until $H > T$. Finally, the logical labels to the labels in $\mathcal{Y}^+$ are set to 1, and other logical labels are set to 0. In our experiments, $T = 0.5$.

5.1.1. Datasets

There are in total one artificial dataset and 13 real-world label distribution datasets[1]. These real-world datasets (Geng, 2016) collected from biological experiments on the yeast genes, facial expression images, natural scene images and movies, respectively. Some basic statistics about these 14 datasets are given in Table 1.

[1]http://palm.seu.edu.cn/xgeng/LDL/index.htm

Table 1. Statistics of the 14 datasets used in the recovery experiment

No.	Dataset	#Examples	#Features	#Labels
1	Artificial (Ar)	2601	3	3
2	SJAFFE (SJ)	213	243	6
3	Yeast-spoem (spoem)	2,465	24	2
4	Yeast-spo5 (spo5)	2,465	24	3
5	Yeast-dtt (dtt)	2,465	24	4
6	Yeast-cold (cold)	2,465	24	4
7	Yeast-heat (heat)	2,465	24	6
8	Yeast-spo (spo)	2,465	24	6
9	Yeast-diau (diau)	2,465	24	7
10	Yeast-elu (elu)	2,465	24	14
11	Yeast-cdc (cdc)	2,465	24	15
12	Yeast-alpha (alpha)	2,465	24	18
13	SBU_3DFE (3DFE)	2,500	243	6
14	Movie (Mov)	7,755	1,869	5

Table 2. The distribution distance/similarity measures

Measure	**Formula**
Chebyshev ↓	$Dis_1(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \max_j \lvert d_j - \hat{d}_j \rvert$
Clark ↓	$Dis_2(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \sqrt{\sum_{j=1}^{c} \frac{(d_j - \hat{d}_j)^2}{(d_j + \hat{d}_j)^2}}$
Canberra ↓	$Dis_3(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \sum_{j=1}^{c} \frac{\lvert d_j - \hat{d}_j \rvert}{d_j + \hat{d}_j}$
Kullback-Leibler ↓	$Dis_4(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \sum_{j=1}^{c} d_j \ln \frac{d_j}{\hat{d}_j}$
cosine ↑	$Sim_1(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \frac{\sum_{j=1}^{c} d_j \hat{d}_j}{\sqrt{\sum_{j=1}^{c} d_j^2} \sqrt{\sum_{j=1}^{c} \hat{d}_j^2}}$
intersection ↑	$Sim_2(\boldsymbol{d}, \hat{\boldsymbol{d}}) = \sum_{j=1}^{c} \min(d_j, \hat{d}_j)$

The artificial dataset is generated to show in a visual way whether the LE algorithms can recover the label distributions from the logical labels. In this dataset, the instance $\boldsymbol{x}$ is of three-dimensional and there are three labels. In order to show the results of LE algorithms in a direct and visual way, the examples of the toy dataset are selected from a certain manifold in the feature space. The first two components of the instance $\boldsymbol{x}$, x_1 and x_2, are located at a grid of the interval 0.04 within the range $[-1, 1]$, and there are in total $51 \times 51 = 2601$ instances. The third component x_3 is calculated by

$$x_3 = \sin((x_1 + x_2) \times \pi). \tag{15}$$

The label distribution $\boldsymbol{d} = [d_{\boldsymbol{x}}^{y_1}, d_{\boldsymbol{x}}^{y_2}, d_{\boldsymbol{x}}^{y_3}]$ of $\boldsymbol{x} = [x_1, x_2, x_3]^\top$ is created to deliberately make the description degree of one label depend on those of other labels (Geng, 2016).

5.1.2. Evaluation Measures

The output of LE algorithm is label distribution rather than logical output of clustering or classification, which makes some commonly used measures inapplicable. As suggested in (Geng, 2016), we select six measures , i.e., Chebyshev distance (Cheb), Clark distance (Clark), Canberra metric

(Canber), Kullback-Leibler divergence (KL), cosine coefficient (Cosine) and intersection similarity (Intersec), which belong to the Minkowski family, the χ^2 family, the L_1 family, the Shannon's entropy family, the inner product family, and the intersection family, respectively. The first four are distance measures and the last two are similarity measures.

Suppose the real label distribution is $\boldsymbol{d} = [d_1, d_2, ..., d_c]$, the predicted label distribution is $\hat{\boldsymbol{d}} = [\hat{d}_1, \hat{d}_2, ..., \hat{d}_c]$, then the formulae of the six measures are summarized in Table 2, where the "$\downarrow$" after the distance measures indicates "the smaller the better", and the "$\uparrow$" after the similarity measures indicates "the larger the better". Considering that the selected measures all come from different families, the selected measures are significantly different in both syntax and semantics.

5.1.3. Methodology

The five LE algorithms, i.e., FCM (Gayar et al., 2006), KM (Jiang et al., 2006), LP (Li et al., 2015), ML (Hou et al., 2016), GLLE (Xu et al., 2018), and our LEVI are all applied to the 14 real-world datasets shown in Table 1. For each compared algorithm, we adopt the suggested configuration in their literature, i.e., the parameter α in LP is set to 0.5, the number of neighbors K for ML is set to $c + 1$, the parameter β in FCM is set to 2, and the kernel function in KM is Gaussian kernel. For GLLE, the parameter λ_1 and λ_2 are chosen among $\{10^{-2}, 10^{-1}, ..., 100\}$, and the number of neighbors K is set to $c+1$. The kernel function in GLLE is Gaussian kernel. For LEVI, the MLPs are constructed with three hidden layers, each with 500 hidden units and softplus activation functions.

5.1.4. Recovery Performance

In order to visually show the results of the LE algorithms on the artificial dataset, the description degrees of the three labels are regarded as the three color channels of the RGB color space, respectively. In this way, the color of a point in the feature space will visually represent its label distribution. Thus, the label distribution recovered by the LE algorithms can be compared with the ground-truth label distribution through observing the color patterns on the manifold. For easier comparison, the images are visually enhanced by applying a decorrelation stretch process. The results are shown in Fig. 3. It can be seen that LEVI recovers almost identical color patterns with the ground-truth. GLLE, LP, ML can also recover similar color patterns with the ground-truth. However, FCM, KM fails to obtain a reasonable result.

For quantitative analysis, table 3 tabulates the results of the five LE algorithms on all real-world the datasets, and the best performance on each dataset is highlighted by boldface. For each evaluation metric, $\downarrow$ indicates the smaller the better while $\uparrow$ indicates the larger the better. Note that since each LE algorithm only runs once, there is no record of standard deviation. we can find that our method achieves optimal average rank in terms of all the six evaluation metrics.

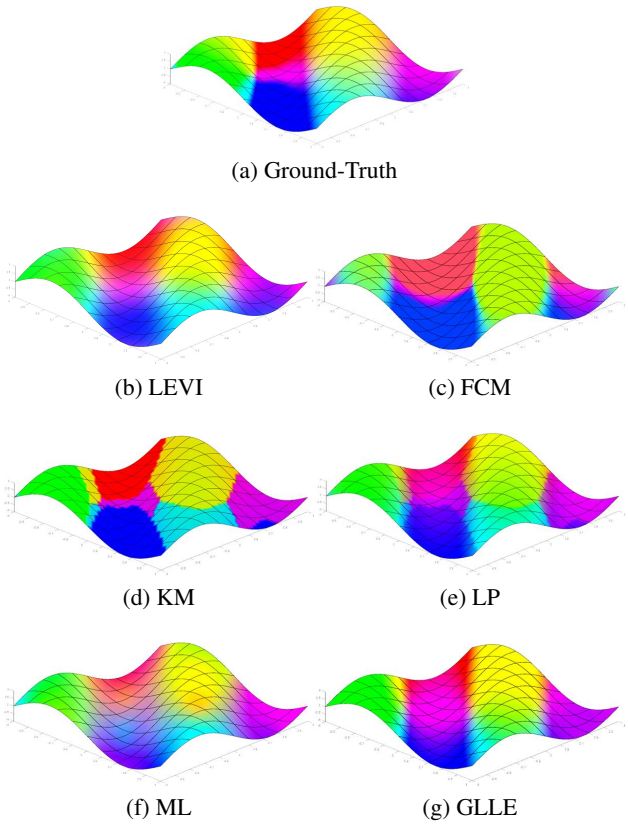

Figure 3. Comparison between the ground-truth and recovered label distributions (regarded as RGB colors) on the artificial manifold.

5.2. Predictive Experiment

In this experiment, the effective performance of LEVI for MLL prediction can be validated.

5.2.1. Datasets

There are ten MLL datasets[2] used in the experiments. Some basic statistics about these datasets are given in Table 4. The MLL datasets cover a broad range of cases with diversified multi-label properties and thus serve as a solid basis for thorough comparative studies.

5.2.2. Evaluation Measures

Five widely-used MLL evaluation metrics are selected in this experiment, i.e., *Hamming loss*, *One-error*, *Coverage*, *Ranking loss* and *Average precision* (Zhang & Zhou, 2014).

[2] mulan.sourceforge.net/datasets.html

Table 3. Recovery results (value(rank)) evaluated by six LDL measures

Comparing algorithm	SJ	spoem	spo5	dtt	cold	heat	spo	diau	elu	cdc	alpha	3DFE	Mov	Avg. Rank
							Cheb ↓							
FCM	0.132(4)	0.233(4)	0.162(4)	0.097(3)	0.141(4)	0.169(5)	0.130(4)	0.124(4)	0.052(4)	0.051(4)	0.044(4)	0.135(4)	0.230(5)	4.08
KM	0.214(6)	0.408(6)	0.277(6)	0.257(6)	0.252(6)	0.175(6)	0.175(6)	0.152(6)	0.078(6)	0.076(6)	0.063(6)	0.238(6)	0.234(6)	6.00
LP	0.107(3)	0.163(3)	0.114(3)	0.128(4)	0.137(3)	0.086(3)	0.090(3)	0.099(3)	0.044(3)	0.042(3)	0.040(3)	0.123(2)	0.161(3)	3.00
ML	0.186(5)	0.403(5)	0.273(5)	0.244(5)	0.242(5)	0.165(4)	0.171(5)	0.148(5)	0.072(5)	0.071(5)	0.057(5)	0.233(5)	0.164(4)	4.85
GLLE	0.087(2)	0.088(2)	0.099(2)	0.052(2)	0.066(2)	0.049(2)	0.062(2)	0.053(2)	0.023(2)	0.022(2)	0.020(2)	0.126(3)	0.122(2)	2.08
LEVI	**0.073(1)**	**0.063(1)**	**0.067(1)**	**0.051(1)**	**0.051(1)**	**0.033(1)**	**0.045(1)**	**0.033(1)**	**0.017(1)**	**0.015(1)**	**0.013(1)**	**0.092(1)**	**0.109(1)**	1.00
							Clark ↓							
FCM	0.522(4)	0.401(4)	0.395(4)	0.329(3)	0.433(3)	0.580(4)	0.520(3)	0.838(4)	0.579(3)	0.739(3)	0.821(3)	0.482(3)	0.859(3)	3.38
KM	1.874(6)	1.028(6)	1.059(6)	1.477(6)	1.472(6)	1.802(6)	1.811(6)	1.886(6)	2.768(6)	2.885(6)	3.153(6)	1.907(6)	1.766(6)	6.00
LP	0.502(3)	0.272(3)	0.274(3)	0.499(4)	0.503(4)	0.568(3)	0.558(4)	0.788(3)	0.973(4)	1.014(4)	1.185(4)	0.580(4)	0.913(4)	3.62
ML	1.519(5)	1.004(5)	1.036(5)	1.446(5)	1.440(5)	1.764(5)	1.768(5)	1.844(5)	2.711(5)	2.825(5)	3.088(5)	1.848(5)	1.140(5)	5.00
GLLE	0.377(2)	0.132(2)	0.197(2)	0.143(2)	0.176(2)	0.213(2)	0.266(2)	0.296(2)	0.295(2)	0.306(2)	0.337(2)	0.391(2)	0.569(2)	2.00
LEVI	**0.285(1)**	**0.098(1)**	**0.136(1)**	**0.140(1)**	**0.140(1)**	**0.147(1)**	**0.187(1)**	**0.191(1)**	**0.222(1)**	**0.209(1)**	**0.219(1)**	**0.304(1)**	**0.548(1)**	1.00
							Canber ↓							
FCM	1.081(4)	0.534(4)	0.563(4)	0.501(3)	0.734(3)	1.157(3)	0.998(3)	1.895(4)	1.689(3)	2.415(3)	2.883(3)	1.020(3)	1.664(3)	3.31
KM	4.010(6)	1.253(6)	1.382(6)	2.594(6)	2.566(6)	3.849(6)	3.854(6)	4.261(6)	9.110(6)	9.875(6)	11.809(6)	4.121(6)	3.444(6)	6.00
LP	1.064(3)	0.365(3)	0.401(3)	0.941(4)	0.924(4)	1.293(4)	1.231(4)	1.748(3)	3.381(4)	3.644(4)	4.544(4)	1.245(4)	1.720(4)	3.69
ML	3.138(5)	1.226(5)	1.355(5)	2.549(5)	2.519(5)	3.779(5)	3.772(5)	4.180(5)	8.949(5)	9.695(5)	11.603(5)	4.001(5)	1.934(5)	5.00
GLLE	0.781(2)	0.183(2)	0.305(2)	0.248(2)	0.305(2)	0.430(2)	0.548(2)	0.671(2)	0.902(2)	0.959(2)	1.134(2)	0.828(2)	1.045(2)	2.00
LEVI	**0.587(1)**	**0.135(1)**	**0.208(1)**	**0.247(1)**	**0.243(1)**	**0.295(1)**	**0.372(1)**	**0.421(1)**	**0.674(1)**	**0.642(1)**	**0.732(1)**	**0.635(1)**	**0.968(1)**	1.00
							KL ↓							
FCM	0.107(4)	0.208(4)	0.123(4)	0.065(3)	0.113(4)	0.147(4)	0.110(4)	0.159(4)	0.059(3)	0.091(3)	0.100(3)	0.094(3)	0.381(5)	3.69
KM	0.558(6)	0.531(6)	0.334(6)	0.617(6)	0.586(6)	0.586(6)	0.562(6)	0.538(6)	0.617(6)	0.630(6)	0.630(6)	0.603(6)	0.452(6)	6.00
LP	0.077(3)	0.067(3)	0.042(3)	0.103(4)	0.103(3)	0.089(3)	0.084(3)	0.127(3)	0.109(4)	0.111(4)	0.121(4)	0.105(4)	0.177(3)	3.38
ML	0.391(5)	0.503(5)	0.317(5)	0.586(5)	0.556(5)	0.556(5)	0.532(5)	0.509(5)	0.589(5)	0.601(5)	0.602(5)	0.565(5)	0.218(4)	4.92
GLLE	0.050(2)	0.027(2)	0.034(2)	0.013(2)	0.019(2)	0.017(2)	0.029(2)	0.027(2)	0.013(2)	0.014(2)	0.013(2)	0.069(2)	0.123(2)	2.00
LEVI	**0.031(1)**	**0.013(1)**	**0.015(1)**	**0.011(1)**	**0.011(1)**	**0.008(1)**	**0.014(1)**	**0.011(1)**	**0.007(1)**	**0.006(1)**	**0.006(1)**	**0.042(1)**	**0.081(1)**	1.00
							Cosine ↑							
FCM	0.906(4)	0.878(4)	0.922(4)	0.959(3)	0.922(4)	0.883(4)	0.909(4)	0.882(4)	0.950(3)	0.929(3)	0.922(3)	0.912(4)	0.773(6)	3.85
KM	0.827(6)	0.812(6)	0.882(6)	0.759(6)	0.779(6)	0.779(6)	0.800(6)	0.799(6)	0.758(6)	0.754(6)	0.751(6)	0.812(6)	0.880(5)	5.92
LP	0.941(3)	0.950(3)	0.969(3)	0.921(4)	0.925(3)	0.932(3)	0.939(3)	0.915(3)	0.918(4)	0.916(4)	0.911(4)	0.922(3)	0.929(3)	3.31
ML	0.857(5)	0.815(5)	0.884(5)	0.763(5)	0.784(5)	0.783(5)	0.803(5)	0.803(5)	0.763(5)	0.759(5)	0.756(5)	0.815(5)	0.919(4)	4.92
GLLE	0.958(2)	0.978(2)	0.971(2)	0.988(2)	0.982(2)	0.984(2)	0.974(2)	0.975(2)	0.987(2)	0.987(2)	0.987(2)	0.927(2)	0.936(2)	2.00
LEVI	**0.970(1)**	**0.990(1)**	**0.987(1)**	**0.990(1)**	**0.990(1)**	**0.992(1)**	**0.988(1)**	**0.990(1)**	**0.993(1)**	**0.994(1)**	**0.995(1)**	**0.957(1)**	**0.955(1)**	1.00
							Intersec ↑							
FCM	0.821(4)	0.767(4)	0.838(4)	0.894(3)	0.833(3)	0.807(3)	0.836(3)	0.760(4)	0.883(3)	0.847(3)	0.844(3)	0.827(3)	0.677(5)	3.46
KM	0.593(6)	0.592(6)	0.724(6)	0.541(6)	0.559(6)	0.559(6)	0.575(6)	0.588(6)	0.539(6)	0.533(6)	0.532(6)	0.579(6)	0.649(6)	6.00
LP	0.837(3)	0.837(3)	0.886(3)	0.786(4)	0.794(4)	0.805(4)	0.819(4)	0.788(3)	0.782(4)	0.779(4)	0.774(4)	0.810(4)	0.778(4)	3.69
ML	0.661(5)	0.597(5)	0.727(5)	0.546(5)	0.565(5)	0.564(5)	0.580(5)	0.593(5)	0.544(5)	0.538(5)	0.537(5)	0.587(5)	0.779(3)	4.85
GLLE	0.872(2)	0.912(2)	0.901(2)	**0.939(1)**	0.924(2)	0.929(2)	0.909(2)	0.906(2)	0.936(2)	0.937(2)	0.938(2)	0.850(2)	0.831(2)	1.92
LEVI	**0.899(1)**	**0.937(1)**	**0.933(1)**	**0.939(1)**	**0.940(1)**	**0.952(1)**	**0.940(1)**	**0.942(1)**	**0.952(1)**	**0.958(1)**	**0.960(1)**	**0.882(1)**	**0.850(1)**	1.00

Table 4. Statistics of the 10 datasets used in MLL predictive experiment

No.	Dataset	#Examples	#Features	#Labels
1	cal500	502	68	174
2	emotion	593	72	6
3	medical	978	1,449	45
4	llog	1,460	1,004	75
5	enron	1,702	1,001	53
6	msra	1,868	898	19
7	image	2,000	294	5
8	scene	2,407	294	5
9	slashdot	3,782	1,079	22
10	corel5k	5,000	499	374

Note that for all the five multi-label metrics, their values vary between [0,1]. Furthermore, for average precision, the *larger* the values the better the performance; While for the other four metrics, the *smaller* the values the better the performance. These metrics serve as good indicators for comprehensive comparative studies as they evaluate the performance of the learned models from various aspects.

5.2.3. Methodology

In this paper, we choose to compare the performance of LEVI against four well established multi-label learning algorithms, including Binary Relevance (BR) (Boutell et al., 2004), Calibrated Label Ranking (CLR) (Fürnkranz et al., 2008), Ensemble of Classifier Chains (ECC) (Read et al., 2011), Random k-labelsets (RAKEL) (Tsoumakas et al., 2011). For ECC, the ensemble size is set to 30. For RAKEL, the ensemble size is set to be $2q$ with $k = 3$ as suggested in the literature (Tsoumakas et al., 2011). Note that some work (Li et al., 2015; Hou et al., 2016; Xu et al., 2019) validate the effectiveness of LP, ML and GLLE in MLL, LEVI is also compared with them. In addition, a deep model (MLP which has the same structure as the encoder of LEVI) trained with logical labels is compared. For each compared algorithm, we adopt the suggested configuration in their literature. For the predictive model in LEVI, the parameters C_1 and C_2 are set to 1 and 10.

Table 5. Predictive performance of each algorithm (mean±std(rank)) measured by five MLL measures.

Comparing algorithm	Ranking-loss ↓										Avg.
	cal500	emotions	medical	llog	enron	image	scene	msra	slashdot	corel5k	Rank
LEVI	**0.177±0.002(1)**	0.192±0.008(2)	**0.024±0.004(1)**	0.154±0.005(3)	**0.080±0.003(1)**	**0.142±0.006(1)**	**0.062±0.004(1)**	**0.126±0.010(1)**	**0.098±0.002(1)**	0.118±0.002(3)	1.50
GLLE	0.179±0.002(2)	0.199±0.011(3)	0.025±0.004(2)	0.141±0.008(2)	0.085±0.002(5)	0.165±0.006(4)	0.073±0.004(4)	0.143±0.009(5)	0.098±0.003(2)	0.137±0.003(6)	3.50
MLP	0.182±0.004(3)	**0.185±0.009(1)**	0.027±0.004(4)	**0.129±0.004(1)**	0.082±0.005(2)	0.181±0.015(5)	0.091±0.003(5)	0.141±0.009(4)	0.128±0.003(6)	0.134±0.004(5)	3.60
LP	0.190±0.002(4)	0.213±0.009(5)	0.030±0.006(5)	0.180±0.007(6)	0.083±0.002(3)	0.162±0.006(3)	0.070±0.004(3)	0.138±0.012(3)	0.100±0.003(3)	0.123±0.002(4)	3.90
ML	0.190±0.002(4)	0.205±0.006(4)	0.026±0.005(3)	0.156±0.006(5)	0.083±0.002(3)	0.160±0.005(2)	0.069±0.004(2)	0.136±0.012(2)	0.104±0.004(4)	0.117±0.002(2)	3.10
BR	0.258±0.003(8)	0.233±0.016(8)	0.091±0.005(7)	0.328±0.007(8)	0.312±0.009(9)	0.314±0.014(9)	0.229±0.010(9)	0.368±0.021(9)	0.240±0.008(8)	0.416±0.003(8)	8.30
CLR	0.239±0.026(7)	0.222±0.014(6)	0.123±0.026(9)	0.190±0.015(7)	0.089±0.002(6)	0.294±0.009(7)	0.127±0.003(6)	0.288±0.018(7)	0.260±0.007(9)	**0.114±0.002(1)**	6.50
ECC	0.205±0.004(6)	0.227±0.017(7)	0.032±0.007(6)	0.154±0.009(4)	0.120±0.004(7)	0.276±0.005(6)	0.151±0.005(7)	0.332±0.047(8)	0.123±0.004(5)	0.292±0.003(7)	6.30
RAKEL	0.444±0.005(9)	0.254±0.020(9)	0.095±0.033(8)	0.412±0.010(9)	0.241±0.005(8)	0.311±0.010(8)	0.205±0.008(8)	0.223±0.075(6)	0.190±0.005(7)	0.627±0.004(9)	8.10
Comparing algorithm	One-error ↓										Avg.
	cal500	emotions	medical	llog	enron	image	scene	msra	slashdot	corel5k	Rank
LEVI	**0.116±0.014(1)**	**0.310±0.017(1)**	**0.155±0.014(1)**	**0.738±0.023(1)**	**0.220±0.009(1)**	**0.271±0.009(1)**	**0.193±0.008(1)**	**0.049±0.015(1)**	**0.383±0.007(1)**	**0.658±0.009(1)**	1.00
GLLE	**0.116±0.014(1)**	0.321±0.021(3)	0.181±0.016(4)	0.762±0.025(4)	0.232±0.011(2)	0.312±0.017(3)	0.224±0.008(4)	0.059±0.015(4)	0.440±0.013(3)	0.665±0.009(2)	3.00
MLP	0.125±0.015(3)	0.315±0.018(2)	0.163±0.013(2)	0.753±0.008(3)	0.236±0.014(3)	0.331±0.020(5)	0.259±0.009(5)	0.057±0.021(2)	0.429±0.009(2)	0.685±0.009(5)	3.20
LP	0.136±0.008(5)	0.342±0.019(5)	0.189±0.021(6)	0.769±0.017(5)	0.238±0.014(5)	0.313±0.009(4)	0.220±0.007(3)	0.060±0.015(5)	0.464±0.014(5)	0.666±0.008(3)	4.60
ML	0.135±0.011(4)	0.336±0.015(4)	0.172±0.015(3)	0.745±0.013(2)	0.237±0.015(4)	0.310±0.010(2)	0.219±0.006(2)	0.058±0.016(3)	0.478±0.015(6)	0.681±0.009(4)	3.40
BR	0.921±0.025(9)	0.375±0.027(8)	0.297±0.036(8)	0.884±0.011(8)	0.648±0.019(9)	0.538±0.019(9)	0.475±0.014(9)	0.464±0.032(9)	0.734±0.017(8)	0.919±0.006(9)	8.60
CLR	0.331±0.111(8)	0.356±0.030(7)	0.688±0.143(9)	0.900±0.019(9)	0.376±0.017(6)	0.514±0.014(7)	0.371±0.008(6)	0.312±0.085(7)	0.979±0.003(9)	0.721±0.007(7)	7.50
ECC	0.191±0.021(6)	0.353±0.040(6)	0.182±0.019(5)	0.785±0.009(6)	0.424±0.013(8)	0.486±0.018(6)	0.373±0.008(7)	0.420±0.105(8)	0.481±0.014(7)	0.699±0.006(6)	6.50
RAKEL	0.286±0.039(7)	0.392±0.035(9)	0.208±0.071(7)	0.838±0.014(7)	0.412±0.016(7)	0.515±0.017(8)	0.444±0.012(8)	0.302±0.103(6)	0.453±0.005(4)	0.819±0.010(8)	7.10
Comparing algorithm	Coverage ↓										Avg.
	cal500	emotions	medical	llog	enron	image	scene	msra	slashdot	corel5k	Rank
LEVI	**0.745±0.007(1)**	0.320±0.009(2)	0.038±0.007(2)	0.158±0.005(2)	0.236±0.007(4)	**0.167±0.006(1)**	**0.066±0.003(1)**	**0.529±0.019(1)**	0.115±0.002(3)	0.278±0.004(2)	1.90
GLLE	0.747±0.006(2)	0.330±0.010(3)	**0.038±0.006(1)**	**0.147±0.009(1)**	0.247±0.007(6)	0.186±0.006(4)	0.075±0.004(4)	0.560±0.013(5)	**0.113±0.003(1)**	0.333±0.007(6)	3.30
MLP	0.749±0.007(3)	**0.308±0.008(1)**	0.040±0.005(3)	0.164±0.005(4)	**0.230±0.009(1)**	0.197±0.012(5)	0.089±0.003(5)	0.543±0.014(2)	0.147±0.003(6)	0.284±0.004(4)	3.40
LP	0.786±0.007(4)	0.339±0.009(5)	0.045±0.009(5)	0.184±0.009(5)	0.236±0.004(2)	0.184±0.006(3)	0.072±0.004(3)	0.551±0.017(4)	0.114±0.004(2)	0.297±0.005(5)	3.80
ML	0.787±0.007(5)	0.330±0.010(3)	0.041±0.008(4)	0.159±0.008(3)	0.236±0.004(2)	0.182±0.006(2)	0.071±0.004(2)	0.549±0.017(3)	0.118±0.004(4)	0.280±0.005(3)	3.10
BR	0.852±0.014(8)	0.363±0.015(8)	0.118±0.007(8)	0.377±0.008(8)	0.601±0.014(9)	0.301±0.012(9)	0.207±0.009(9)	0.759±0.018(9)	0.259±0.009(8)	0.758±0.003(8)	8.40
CLR	0.794±0.010(7)	0.351±0.016(6)	0.143±0.030(9)	0.225±0.016(7)	0.243±0.006(5)	0.286±0.008(7)	0.120±0.007(6)	0.720±0.023(7)	0.272±0.007(9)	**0.267±0.004(1)**	6.40
ECC	0.788±0.008(6)	0.356±0.013(7)	0.048±0.009(6)	0.192±0.010(6)	0.300±0.009(7)	0.272±0.005(6)	0.141±0.004(7)	0.743±0.033(8)	0.139±0.004(5)	0.562±0.007(7)	6.50
RAKEL	0.971±0.001(9)	0.381±0.019(9)	0.117±0.040(7)	0.459±0.011(9)	0.523±0.008(8)	0.298±0.010(8)	0.186±0.006(8)	0.628±0.210(6)	0.212±0.005(7)	0.886±0.004(9)	8.00
Comparing algorithm	Hamming-loss ↓										Avg.
	cal500	emotions	medical	llog	enron	image	scene	msra	slashdot	corel5k	Rank
LEVI	**0.137±0.002(1)**	0.224±0.008(2)	0.012±0.001(2)	**0.015±0.000(1)**	**0.047±0.001(1)**	**0.157±0.003(1)**	**0.080±0.002(1)**	**0.182±0.009(1)**	**0.039±0.001(1)**	**0.009±0.000(1)**	1.20
GLLE	0.140±0.002(3)	0.225±0.007(3)	0.013±0.001(4)	0.025±0.007(8)	0.052±0.001(5)	0.218±0.006(5)	0.142±0.005(6)	0.200±0.006(5)	0.042±0.001(2)	0.012±0.000(6)	4.70
MLP	0.141±0.002(5)	**0.224±0.006(1)**	0.012±0.001(2)	**0.015±0.000(1)**	0.048±0.001(2)	0.177±0.009(2)	0.097±0.003(2)	0.199±0.006(4)	0.043±0.001(3)	**0.009±0.000(1)**	2.30
LP	0.143±0.002(6)	0.243±0.005(5)	0.019±0.001(6)	0.015±0.000(2)	0.050±0.001(3)	0.188±0.003(4)	0.103±0.002(4)	0.190±0.010(3)	0.048±0.000(4)	0.009±0.000(2)	3.90
ML	0.140±0.002(3)	0.231±0.008(4)	0.019±0.001(6)	0.015±0.000(3)	0.051±0.001(4)	0.180±0.003(3)	0.099±0.002(3)	0.189±0.010(2)	0.048±0.000(4)	0.009±0.000(3)	3.50
BR	0.214±0.004(9)	0.265±0.013(7)	0.022±0.003(8)	0.052±0.003(9)	0.105±0.003(9)	0.287±0.008(8)	0.184±0.005(9)	0.404±0.037(9)	0.130±0.003(9)	0.027±0.000(9)	8.60
CLR	0.165±0.005(8)	0.270±0.011(9)	0.024±0.002(9)	0.019±0.002(7)	0.072±0.002(8)	0.305±0.005(9)	0.181±0.004(8)	0.342±0.033(7)	0.058±0.001(8)	0.011±0.001(5)	7.80
ECC	0.146±0.002(7)	0.254±0.013(6)	0.013±0.001(4)	0.016±0.000(5)	0.064±0.001(7)	0.244±0.005(6)	0.133±0.002(5)	0.353±0.037(8)	0.049±0.001(7)	0.015±0.001(8)	6.30
RAKEL	0.138±0.002(2)	0.269±0.011(8)	**0.010±0.003(1)**	0.017±0.001(6)	0.058±0.001(6)	0.286±0.007(7)	0.171±0.005(7)	0.237±0.079(6)	0.048±0.001(6)	0.012±0.001(7)	5.60
Comparing algorithm	Average-precision ↑										Avg.
	cal500	emotions	medical	llog	enron	image	scene	msra	slashdot	corel5k	Rank
LEVI	**0.511±0.004(1)**	0.773±0.008(2)	**0.879±0.014(1)**	**0.367±0.013(1)**	**0.697±0.008(1)**	**0.824±0.005(1)**	**0.887±0.005(1)**	**0.826±0.013(1)**	**0.710±0.005(1)**	**0.297±0.003(1)**	1.10
GLLE	0.501±0.003(3)	0.764±0.011(3)	0.866±0.013(4)	0.353±0.016(2)	0.680±0.005(3)	0.799±0.009(3)	0.869±0.005(4)	0.806±0.011(5)	0.668±0.008(3)	0.285±0.004(3)	3.30
MLP	0.504±0.006(2)	**0.778±0.007(1)**	0.876±0.011(2)	0.332±0.008(6)	0.688±0.010(2)	0.786±0.014(5)	0.844±0.005(5)	0.808±0.013(4)	0.668±0.006(2)	0.283±0.004(4)	3.30
LP	0.492±0.002(5)	0.752±0.010(5)	0.852±0.018(6)	0.339±0.014(5)	0.664±0.006(4)	0.798±0.005(4)	0.872±0.004(3)	0.810±0.016(3)	0.654±0.009(4)	0.293±0.003(2)	4.10
ML	0.497±0.002(4)	0.758±0.006(4)	0.869±0.013(3)	0.350±0.012(3)	0.662±0.007(5)	0.800±0.005(2)	0.873±0.004(2)	0.813±0.015(2)	0.642±0.010(5)	0.279±0.003(5)	3.50
BR	0.300±0.005(9)	0.730±0.015(8)	0.762±0.022(7)	0.215±0.009(7)	0.381±0.009(9)	0.649±0.012(9)	0.692±0.010(9)	0.540±0.015(9)	0.427±0.014(8)	0.123±0.003(8)	8.30
CLR	0.395±0.042(7)	0.742±0.016(6)	0.400±0.062(9)	0.194±0.018(9)	0.610±0.008(6)	0.666±0.008(7)	0.778±0.004(6)	0.624±0.022(6)	0.250±0.007(9)	0.274±0.002(6)	7.10
ECC	0.463±0.006(6)	0.740±0.021(7)	0.860±0.015(5)	0.342±0.009(4)	0.559±0.008(7)	0.685±0.008(6)	0.766±0.005(7)	0.567±0.048(8)	0.628±0.009(6)	0.264±0.003(7)	6.30
RAKEL	0.353±0.006(8)	0.717±0.023(9)	0.700±0.234(8)	0.197±0.013(8)	0.539±0.006(8)	0.661±0.010(8)	0.713±0.008(8)	0.601±0.200(7)	0.617±0.004(7)	0.122±0.004(9)	8.00

5.2.4. PREDICTIVE PERFORMANCE

Table 5 tabulates the results of all the algorithms (LEVI, GLLE, MLP, LP, ML, BR, CLR, ECC and RAKEL) on the ten MLL datasets evaluated by five evaluation metrics, and the best performance on each dataset is highlighted by boldface. For each evaluation metric, ↓ indicates the smaller the better while ↑ indicates the larger the better. All the algorithms are tested via ten-fold cross validation. The ranks are given in the parentheses right after the performance values. The average rank of each algorithm over all the datasets is also calculated and given in the last row of each table.

When looking at the average ranks over all the ten real-world datasets, LEVI achieves rather competitive performance over other algorithms. Besides, the rankings of each LE based algorithm on five measures are higher than the four state-of-the-art MLL algorithms. When compared with the state-of-the-art MLL algorithms, LEVI ranks 1st in 84.0% cases and ranks 2nd in 10.0% cases. Thus, LEVI based MLL algorithm achieves rather superior performance over the state-of-the-art multi-label learning algorithms across all the evaluation measures.

6. Conclusion

Label enhancement can recover the label distributions from the logical labels in the training sets, which reinforces the supervision information in the training sets. By induce the generative model of the label distribution and adopt the variational inference technique, we give a lower bound of the label distribution and propose a novel LE approach called Label Enhancement via Variational Inference (LEVI) to infer the label distributions from the logical labels. Extensive

comparative studies clearly validate the advantage of LEVI against other LE algorithms and the effectiveness of MLL after LE pre-process on the logical-labeled datasets.

Acknowledgements

This research was supported by the National Key Research & Development Plan of China (No. 2017YFB1002801), the National Science Foundation of China (61622203), the Collaborative Innovation Center of Novel Software Technology and Industrialization, and the Collaborative Innovation Center of Wireless Communications Technology.

References

Boutell, M. R., Luo, J., Shen, X., and Brown, C. M. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

Cabral, R. S., De la Torre, F., Costeira, J. P., and Bernardino, A. Matrix completion for multi-label image classification. In *Advances in Neural Information Processing Systems*, pp. 190–198, Granada, Spain, 2011.

Denœux, T. and Zouhal, L. M. Handling possibilistic labels in pattern classification using evidential reasoning. *Fuzzy sets and systems*, 122(3):409–424, 2001.

Fürnkranz, J., Hüllermeier, E., Mencía, E. L., and Brinker, K. Multilabel classification via calibrated label ranking. *Machine Learning*, 73(2):133–153, 2008.

Gao, B.-B., Zhou, H.-Y., Wu, J., and Geng, X. Age estimation using expectation of label distribution learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 712–718, Stockholm, Sweden, 2018.

Gayar, N. E., Schwenker, F., and Palm, G. A study of the robustness of knn classifiers trained using soft labels. In *Proceedings of the 2nd International Conference on Artificial Neural Network in Pattern Recognition*, pp. 67–80, Ulm, Germany, 2006.

Geng, X. Label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 28(7):1734–1748, 2016.

Geng, X. and Luo, L. Multilabel ranking with inconsistent rankers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3742–3747, Columbus, OH, 2014.

Geng, X. and Xia, Y. Head pose estimation based on multivariate label distribution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1837–1842, Columbus, OH, 2014.

Geng, X., Yin, C., and Zhou, Z.-H. Facial age estimation by learning from label distributions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(10): 2401–2412, 2013.

Gibaja, E. and Ventura, S. A tutorial on multilabel learning. *ACM Computing Surveys*, 47(3):1–38, 2015.

Hou, P., Geng, X., and Zhang, M.-L. Multi-label manifold learning. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pp. 1680–1686, Phoenix, AZ, 2016.

Huo, Z. and Geng, X. Ordinal zero-shot learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 1331–1337, Melbourne, Australia, 2017.

Jiang, X., Yi, Z., and Lv, J. C. Fuzzy svm with a new fuzzy membership function. *Neural Computing & Applications*, 15(3-4):268–276, 2006.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Li, Y.-K., Zhang, M.-L., and Geng, X. Leveraging implicit relative labeling-importance information for effective multi-label learning. In *Proceedings of the 15th IEEE International Conference on Data Mining*, pp. 251–260, Atlantic City, NJ, 2015.

Lo, H.-Y., Wang, J.-C., Wang, H.-M., and Lin, S.-D. Cost-sensitive multi-label learning for audio tag annotation and retrieval. *IEEE Transactions on Multimedia*, 13(3): 518–529, 2011.

Quost, B. and Denœux, T. Learning from data with uncertain labels by boosting credal classifiers. In *Proceedings of the 1st ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data*, pp. 38–47, Paris, France, 2009.

Read, J., Pfahringer, B., Holmes, G., and Frank, E. Classifier chains for multi-label classification. *Machine Learning*, 85(3):333, 2011.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Rubin, T. N., Chambers, A., Smyth, P., and Steyvers, M. Statistical topic models for multi-label document classification. *Machine Learning*, 88(1-2):157–208, 2012.

Smyth, P., Fayyad, U., Burl, M., Perona, P., and Baldi, P. Learning with probabilistic supervision. *Computational learning theory and natural learning systems*, 3:163–182, 1995.

Su, K. and Geng, X. Soft facial landmark detection by label distribution learning. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, pp. in press, Honolulu, HI, 2019.

Tsoumakas, G. and Katakis, I. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining*, 3(3):1–13, 2006.

Tsoumakas, G., Katakis, I., and Vlahavas, I. Random k-labelsets for multilabel classification. *IEEE Transactions on Knowledge and Data Engineering*, 23(7):1079–1089, 2011.

Wang, J. and Geng, X. Theoretical analysis of label distribution learning. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, pp. in press, Honolulu, HI, 2019.

Wang, J., Zhao, Y., Wu, X., and Hua, X.-S. A transductive multi-label learning approach for video concept detection. *Pattern Recognition*, 44(10):2274–2286, 2011.

Xu, N., Tao, A., and Geng, X. Label enhancement for label distribution learning. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 2926–2932, Stockholm, Sweden, 2018.

Xu, N., Liu, Y.-P., and Geng, X. Label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, pp. in press, 2019.

Zhang, M.-L. and Zhou, Z.-H. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2014.

Zhou, D., Zhou, Y., Zhang, X., Zhao, Q., and Geng, X. Emotion distribution learning from texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 638–647, Austin, TX, 2016.

Zhu, X., Lafferty, J., and Rosenfeld, R. *Semi-supervised learning with graphs*. Carnegie Mellon University, language technologies institute, school of computer science, 2005.