Label Enhancement via Joint Implicit Representation Clustering

Yunan Lu[1] , **Weiwei Li**[2*] and **Xiuyi Jia**[1]

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, China
[2]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, China

luyn@njust.edu.cn, liweiwei@nuaa.edu.cn, jiaxy@njust.edu.cn

Abstract

Label distribution is an effective label form for representing label polysemy (i.e., the cases where an instance can be described by multiple labels simultaneously). However, the expensive annotating cost of label distributions limits their application to a wide range of practical tasks. Hence, LE (label enhancement) techniques have been extensively studied to solve this problem. Existing LE algorithms mostly estimate label distributions by the instance relation or label relation. However, they suffer from biased instance relations, limited model capabilities, or suboptimal local label correlations. Therefore, in this paper, we propose a deep generative model called JRC to simultaneously learn and cluster the joint implicit representations of both features and labels, which can be used to improve any existing LE algorithm involving the instance relation or local label correlations. Besides, we develop a novel label distribution recovery module, and then integrate it with JRC model, thus constituting a novel generative label enhancement model that utilizes the learned joint implicit representations and instance clusters in a principled manner. Finally, extensive experiments validate our proposal.

1 Introduction

Label polysemy, i.e., the cases where an instance can be described by multiple labels simultaneously, is common in practical machine learning tasks. Label distribution is an effective label form to describe label polysemy, where each label is assigned a real value to indicate how much this label describes the instance. Since label distribution provides fine-grained information about label polysemy, it has been widely applied in many practical tasks, such as emotion analysis [Jia *et al.*, 2019; He and Jin, 2019; Peng *et al.*, 2015], age estimation [Gao *et al.*, 2018; Wen *et al.*, 2020; Shen *et al.*, 2021], and so on.

A popular topic regarding label distribution is how to predict label distributions for unseen instances. LDL (label distribution learning) [Geng, 2016] is an effective learning paradigm for solving this problem. However, LDL requires the training instances annotated with label distributions, yet quantifying label distributions is costly and even impractical.

Therefore, a technique called LE (label enhancement) [Xu *et al.*, 2018] was proposed to solve this problem. In general, existing works about LE mostly assume that instances annotated with simple labels (such as logical labels [Xu *et al.*, 2021] or multi-label rankings [Lu and Jia, 2022]) are available, then label distributions can be recovered by mining the *instance relation* or the *label relation*, and finally any LDL algorithm can be trained to predict label distributions for unseen instances. For example, in terms of the instance relation, some algorithms [Hou *et al.*, 2016; Zheng *et al.*, 2023; Wen *et al.*, 2021] represent the feature vector of each instance as a linear combination of the feature vectors of its neighbors and then make the instances retain this combination in the label distribution space. There are also some algorithms [Xu *et al.*, 2018; Zhang *et al.*, 2021; Xu *et al.*, 2023] that construct an instance graph (whose edges encode the similarities of instances in the feature space), and use the Laplacian matrix of this graph to regularize the label distributions. In terms of label relation, N-LDL [Luo *et al.*, 2021] exploits a global label correlation, i.e., all instances share a common label correlation pattern. Besides, more algorithms [Lv *et al.*, 2019; Xu *et al.*, 2021; Jia *et al.*, 2023] leverage local label correlations, i.e., instances from different clusters are regularized by different label correlation patterns.

Overall, above algorithms mostly rely on two processes: mining the instance relation, and clustering instances (for capturing more accurate local label correlations). In terms of these two processes, existing algorithms suffer from the following three problems: 1) These algorithms merely mine the instance relation in the feature space; however, due to the semantic differences between features and labels, regularizing label distributions with the instance relation entirely dependent on features can lead to considerable errors. Moreover, simple labels are more semantically consistent with label distributions, while their potential instance relation is ignored by most algorithms. As shown in Figure 1, the overall error of using only features (blue part) significantly exceeds that of using both features and simple labels (orange part). 2) Existing approaches for mining instance relation are mainly tailored to tabular features; however, real-world features are often diverse (e.g., images, texts, or graphs), and it is unreliable

*Corresponding author is Weiwei Li.

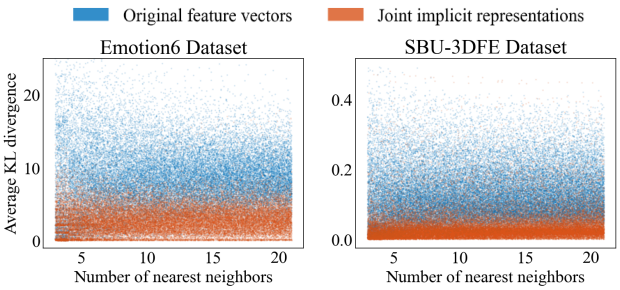


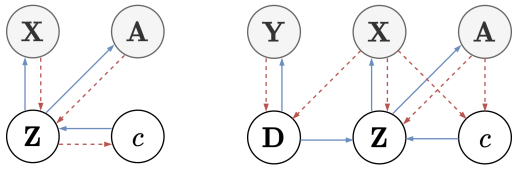
Figure 1: Error distributions. The error is quantified by the average KL (Kullback-Leibler) divergence of the label distribution between an instance and its neighbors. The blue part of each figure indicates that the neighbors are determined by the original feature vectors, which can thus represent the quality of the instance relations that only the features are used. The orange part indicates that both features and labels are used.

to naively apply these approaches to diverse features. 3) Although some algorithms try to extract richer representations by the kernel method, they typically treat instance clustering as independent of learning representations, which may lead to suboptimal results.

Therefore, we propose a deep generative model called JRC to simultaneously learn and cluster the joint implicit representations of both features and simple labels, which can be used to improve any existing LE algorithm that involves mining the instance relation or local label correlations. As shown in Figure 2(a), we treat the joint implicit representations as low-dimensional latent vectors $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^n$, which can probabilistically generate the feature matrix $\mathbf{X}$ and the simple-label-based instance graph $\mathbf{A}$, then model the variational posterior of $\mathbf{z}_i$ by the $\mathbf{A}$-based graph convolutional network, and finally employ the variational Bayes to infer the implicit representation. To encourage the model to adaptively explore instance clusters, inspired by VaDE [Jiang *et al.*, 2017], we assume that $\mathbf{z}_i$ is generated from a Gaussian mixture prior, i.e., the cluster to which $\mathbf{z}_i$ belongs is dominated by a categorical indicator c. Besides, we propose a novel label distribution recovery module which is then integrated with JRC, thus constituting a novel generative label enhancement model called LEIC (i.e., **L**abel **E**nhancement model via joint **I**mplicit representation **C**lustering). As shown in Figure 2(b), in order to utilize the joint implicit representations, we additionally assume that the Gaussian priors of the implicit representations in JRC are conditionally controlled by the label distributions $\mathbf{D}$; in order to utilize local label correlations, we use the cluster posteriors to reweight the instances and construct multiple cluster-specific label correlation graphs which can be then incorporated into the variational posteriors of label distributions via graph convolutional networks. Besides, following LEVI [Xu *et al.*, 2020] and GLEMR [Lu *et al.*, 2023], we also assume that label distributions can generate simple labels $\mathbf{Y}$ due to their strong consistency. Finally, we conduct extensive experiments to show the effectiveness of our proposal.

Our main contributions can be summarized as:

- We propose a deep generative model to learn the joint implicit representations of both features and simple labels, which facilitates the acquisition of more accurate instance relations and enables dealing with various forms of features.
- We incorporate Gaussian mixture clustering and implicit representation learning into a unified framework, which encourages instance clustering and representation learning to accommodate each other and thus facilitates the discovery of more accurate local label correlations.
- We propose a novel generative LE model to recover label distributions, which utilizes the instance relation and local label correlations in a principled way and performs the processes of learning and clustering joint representations and recovering label distributions end-to-end.



(a) JRC (b) LEIC

Figure 2: Diagrams of JRC and LEIC. The solid and dashed directed lines denote the generation and inference processes, respectively.

2 Related Work

Traditional label enhancement considers how to recover label distributions from logical labels. Recently, there was also work that considers recovering label distributions from multi-label rankings. Therefore, label enhancement can be defined in a general sense as the process of enhancing simple labels into label distributions. Simple labels, either logical labels or multi-label rankings, are mostly enhanced into label distributions by the following ways. On the one hand, the most widely studied is estimating label distributions by mining instance relations. Specifically, some algorithms [El Gayar *et al.*, 2006; Jiang *et al.*, 2006; Wang *et al.*, 2023] find the representative points of each label in the feature space and then estimate the relative importance among the labels based on the distance from each instance point to the representative points. Some algorithms [Xu *et al.*, 2018; Zhang *et al.*, 2021; Xu *et al.*, 2019; Liu *et al.*, 2021b] directly compute pairwise instance distances (or affinities) according to the feature vectors and use them to regularize the label distribution vectors. Some algorithms use the technique of manifold learning [Izenman, 2012; Bengio *et al.*, 2013]; they re-express the feature vector of each instance as a linear combination of the feature vectors of its neighbors and then make the instances retain this combination in the label distribution space. Specifically, some of these algorithms [Hou *et al.*, 2016; Shao *et al.*, 2018; Liu *et al.*, 2021a] do not impose additional constraints on the re-expression coefficients; some algorithms [Wen *et al.*, 2021; Tang *et al.*, 2020; Zheng *et al.*, 2023] assume that the re-expression coefficients matrix are of low rank; some algorithms [Zhang *et al.*, 2018; Lv *et al.*, 2019; Xu *et al.*, 2023] consider these coefficients to be sparse.

On the other hand, label relations were also studied in many label enhancement works. For example, N-LDL [Luo *et al.*, 2021] mines global label correlation, i.e., all instances share a label correlation pattern. Some algorithms [Jia *et al.*, 2023; Xu *et al.*, 2021] mines local label correlation, i.e., instances belonging to the same cluster share a label correlation pattern.

3 Methodology

3.1 Problem Formulation

We deal with the datasets that appear as pairs $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, where $\mathbf{x}_i$ and $\mathbf{y}_i$ denote the i-th vector of features and the i-th vector of simple labels (either logical labels or multi-label rankings are possible), respectively. $\mathbf{X} = [\mathbf{x}_1; \cdots; \mathbf{x}_n]$ and $\mathbf{Y} = [\mathbf{y}_1; \cdots; \mathbf{y}_n]$ denote the feature matrix and simple label matrix, respectively. The goal of LE is to infer latent label distributions $\mathbf{D} = [\mathbf{d}_1; \cdots; \mathbf{d}_n]$ based on $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$.

3.2 JRC Model

Generation Process

Since we use a deep generative model to infer the joint implicit representations of instances, we first describe the process of generating data points.

1. Specify the index of the Gaussian prior from a categorical distribution: $p(\mathbf{c}) = \prod_{i=1}^n \mathrm{Cat}(c_i | k^{-1} \cdot \mathbf{1}_k)$, where $\mathbf{1}_k$ is a k-dimensional all-ones vector.
2. Generate joint implicit representations from the chosen Gaussian: $p(\mathbf{Z}|\mathbf{c}) = \prod_{i=1}^n \mathcal{N}(\mathbf{z}_i | \boldsymbol{\mu}_{c_i}, \mathrm{diag}(\boldsymbol{\sigma}_{c_i}^2))$, where $\boldsymbol{\mu}_{c_i}$ and $\boldsymbol{\sigma}_{c_i}$ are learnable parameters.
3. Generate the observations of feature vectors: $p(\mathbf{X}|\mathbf{Z}) = \prod_{i=1}^n \mathcal{N}(\boldsymbol{\mu}_{\mathbf{z}_i}, \mathrm{diag}(\boldsymbol{\sigma}_{\mathbf{z}_i}^2))$, $[\boldsymbol{\mu}_{\mathbf{z}_i}; \zeta^{-1}(\boldsymbol{\sigma}_{\mathbf{z}_i})] = f(\mathbf{z}_i; \boldsymbol{\theta}_{\mathbf{z}})$, where $f(\mathbf{z}_i; \boldsymbol{\theta}_{\mathbf{z}})$ is given by a neural network with parameters $\boldsymbol{\theta}_{\mathbf{z}}$; $\zeta^{-1}(\cdot)$ is the inverse of softplus function.
4. Generate the adjacency matrix of the instance graph based on $\mathbf{Y}$: $p(\mathbf{A}|\mathbf{Z}) = \prod_{i=1}^n \prod_{j=1}^n \mathrm{Ber}(\mathrm{A}_{ij} | \sigma(\mathbf{z}_i^\top \mathbf{z}_j))$, where the adjacency matrix is obtained by
$$A_{ij} = \mathbb{I}(\mathbf{x}_i \in knn(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in knn(\mathbf{x}_i)),$$
where $\mathbb{I}(\cdot)$ is the indicator function, $knn(\mathbf{x})$ denotes the set of instances near $\mathbf{x}$, and in this paper we consider 20 nearest neighbors, i.e., $|knn(\mathbf{x})| = 20$.

The above generation process will derive the following factorization of the joint density:

$$p(\mathbf{X}, \mathbf{A}, \mathbf{Z}, \mathbf{c}) = p(\mathbf{c})p(\mathbf{Z}|\mathbf{c})p(\mathbf{X}|\mathbf{Z})p(\mathbf{A}|\mathbf{Z}). \tag{1}$$

Variational Inference

Since obtaining the exact posteriors of the latent variables in the generative model is intractable, we approximate them by the variational distributions. As shown in Figure 2(a), we assume that the variational posterior of latent variables can be factorized as:

$$q(\mathbf{c}, \mathbf{Z}|\mathbf{X}, \mathbf{A}) = q(\mathbf{Z}|\mathbf{A}, \mathbf{X})q(\mathbf{c}|\mathbf{Z}). \tag{2}$$

Here, to fully exploit the instance relation from simple labels, we use the graph convolutional network to parameterize $q(\mathbf{Z}|\mathbf{A}, \mathbf{X})$, i.e.,

$$\begin{aligned} &q(\mathbf{Z}|\mathbf{X}, \mathbf{A}) = \textstyle\prod_{i=1}^n \mathcal{N}(\mathbf{z}_i | \hat{\boldsymbol{\mu}}_{\mathbf{x}_i}, \mathrm{diag}(\hat{\boldsymbol{\sigma}}_{\mathbf{x}_i})), \\ &[\hat{\boldsymbol{\mu}}_{\mathbf{x}_i}; \zeta^{-1}(\hat{\boldsymbol{\sigma}}_{\mathbf{x}_i})] = g_i(\mathbf{X}, \mathbf{A}; \boldsymbol{\phi}_{\mathbf{x}}), \end{aligned} \tag{3}$$

where $g(\mathbf{X}, \mathbf{A}; \boldsymbol{\phi}_{\mathbf{x}})$ denotes a graph convolutional network with adjacency matrix of $\mathbf{A}$ and learnable parameter of $\boldsymbol{\phi}_{\mathbf{x}}$; $g_i(\mathbf{X}, \mathbf{A}; \boldsymbol{\phi}_{\mathbf{x}})$ is defined as the i-th vector of $g(\mathbf{X}, \mathbf{A}; \boldsymbol{\phi}_{\mathbf{x}})$. Besides, the cluster posterior can be obtained by:

$$q(c_i = j|\mathbf{z}_i) = \frac{p(\mathbf{z}_i|c_i = j)p(c_i = j)}{\sum_{t=1}^k p(\mathbf{z}_i|c_i = t)p(c_i = t)}. \tag{4}$$

We aim to minimize the KL divergence between the variational posterior and the true posterior, which is equivalent to maximizing the ELBO (evidence lower bound). According to the factorizations Equation (1) and Equation (2), the ELBO can be rewritten as:

$$\begin{aligned} &\mathcal{L}_{\mathrm{ELBO}} = \mathbb{E}_{q(\mathbf{Z}|\mathbf{A},\mathbf{X})}[\log p(\mathbf{A}|\mathbf{Z}) + \log p(\mathbf{X}|\mathbf{Z})] - \\ &\mathbb{E}_{q(\mathbf{c}|\mathbf{Z})}[\mathrm{KL}(q(\mathbf{Z}|\mathbf{A}, \mathbf{X}) \| p(\mathbf{Z}|\mathbf{c}))] - \mathrm{KL}(q(\mathbf{c}|\mathbf{Z}) \| p(\mathbf{c})). \end{aligned} \tag{5}$$

In Equation (5), the first term measures the likelihood that the variational posteriors reconstruct the observed data; The second and third terms measure the similarity between the learned variational posteriors and the prior beliefs. In order to optimize Equation (5) using gradient ascent techniques, we turn to SGVB estimator [Kingma and Welling, 2014]. Since $q(\mathbf{z}_i|\mathbf{A}, \mathbf{X})$ is Gaussian, it can be reparameterized by:

$$q(\mathbf{z}_i|\mathbf{A}, \mathbf{X}) = \mathcal{T}(\boldsymbol{\epsilon}_i) \triangleq \hat{\boldsymbol{\mu}}_{\mathbf{x}_i} + \hat{\boldsymbol{\sigma}}_{\mathbf{x}_i} \odot \boldsymbol{\epsilon}_i, \ \ \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}). \tag{6}$$

Then the first term in Equation (5) can be estimated by:

$$\begin{aligned} &\mathbb{E}_{q(\mathbf{Z}|\mathbf{A},\mathbf{X})}[\log p(\mathbf{A}|\mathbf{Z}) + \log p(\mathbf{X}|\mathbf{Z})] \approx L^{-1} \textstyle\sum_{l=1}^L \\ &\log p(\mathbf{A}|[\mathcal{T}(\boldsymbol{\epsilon}_i^{(l)})]_{i=1}^n) + \log p(\mathbf{X}|[\mathcal{T}(\boldsymbol{\epsilon}_i^{(l)})]_{i=1}^n), \end{aligned} \tag{7}$$

where L is the number of Monte Carlo samples, $\boldsymbol{\epsilon}_i^{(l)}$ is a sample from the standard normal distribution. Since $\mathbf{c}$ is discrete, the third term has a closed form, i.e.,

$$\mathrm{KL}(q(\mathbf{c}|\mathbf{Z}) \| p(\mathbf{c})) = \sum_{i=1}^n \sum_{j=1}^k q(c_i = j|\mathbf{z}_i) \log \frac{q(c_i = j|\mathbf{z}_i)}{p(c_i = j)}, \tag{8}$$

and the second term can be rewritten as:

$$\begin{aligned} &\mathbb{E}_{q(\mathbf{c}|\mathbf{Z})}[\mathrm{KL}(q(\mathbf{Z}|\mathbf{A}, \mathbf{X}) \| p(\mathbf{Z}|\mathbf{c}))] = \textstyle\sum_{i=1}^n \sum_{j=1}^k \\ &q(c_i = j|\mathbf{z}_i) \mathrm{KL}(q(\mathbf{z}_i|\mathbf{A}, \mathbf{X}) \| p(\mathbf{z}_i|c_i = j)), \end{aligned} \tag{9}$$

where the KL divergence between two Gaussian distributions, i.e., $\mathrm{KL}(q(\mathbf{z}_i|\mathbf{A}, \mathbf{X}) \| p(\mathbf{z}_i|c_i = j))$, has a closed form [Kingma and Welling, 2014]. Substituting Equation (7), Equation (8) and Equation (9) into Equation (5) yields the optimization objective of JRC.

3.3 LEIC Model

Generation Process

Based on the JRC model, the generation process of LEIC makes two additional assumptions as follows: 1) Parameters of the Gaussian mixture prior of $\mathbf{z}$ are conditionally dominated by label distributions, which follows a Dirichlet prior. 2) Label distributions can probabilistically generate simple labels. According, the joint density can be factorized as:

$$\begin{aligned} &p(\mathbf{Y}, \mathbf{X}, \mathbf{A}, \mathbf{D}, \mathbf{Z}, \mathbf{c}) \\ &= p(\mathbf{c})p(\mathbf{D})p(\mathbf{Z}|\mathbf{c}, \mathbf{D})p(\mathbf{Y}|\mathbf{D})p(\mathbf{X}|\mathbf{Z})p(\mathbf{A}|\mathbf{Z}), \end{aligned} \tag{10}$$

where $p(\mathbf{c})$, $p(\mathbf{X}|\mathbf{Z})$, and $p(\mathbf{A}|\mathbf{Z})$ are the same as in JRC; the prior of label distributions is $p(\mathbf{D}) = \prod_{i=1}^{n} \mathrm{Dir}(\mathbf{d}_i|\mathbf{1}_m)$, where m denotes the number of labels; the conditional distribution $p(\mathbf{Z}|\mathbf{c}, \mathbf{D})$ is defined as:

$$p(\mathbf{Z}|\mathbf{c}, \mathbf{D}) = \prod_{i=1}^{n} \mathcal{N}(\mathbf{z}_i|\boldsymbol{\mu}_{\mathbf{d}_i c_i}, \mathrm{diag}(\boldsymbol{\sigma}^2_{\mathbf{d}_i c_i})), \tag{11}$$

where $[\boldsymbol{\mu}_{\mathbf{d}_i c_i}; \zeta^{-1}(\boldsymbol{\sigma}_{\mathbf{d}_i c_i})] = f(\mathbf{d}_i; \boldsymbol{\theta}_{c_i})$; $f(\mathbf{d}_i; \boldsymbol{\theta}_{c_i})$ is given by a neural network with parameters $\boldsymbol{\theta}_{c_i}$. If the simple labels are logical labels, $p(\mathbf{Y}|\mathbf{D})$ can be a Bernoulli distribution; if the simple labels are multi-label rankings, $p(\mathbf{Y}|\mathbf{D})$ can be a Plackett-Luce distribution [Plackett, 1975].

Variational Inference

Compared to JRC, LEIC additionally needs to infer the posterior of label distributions. We assume the variational posterior of $\mathbf{D}$ as $q(\mathbf{D}|\mathbf{X}, \mathbf{Y}) = \prod_{i=1}^{n} \mathrm{Dir}(\mathbf{d}_i|\hat{\boldsymbol{\alpha}}_i)$, where

$$\hat{\boldsymbol{\alpha}}_i = \tilde{\boldsymbol{G}}_i \cdot \zeta \circ f(\mathbf{x}_i; \boldsymbol{\phi}_d) + \lambda \mathbf{y}_i. \tag{12}$$

In Equation (12), the first term (i.e., $\tilde{\boldsymbol{G}}_i \cdot \zeta \circ f(\mathbf{x}_i; \boldsymbol{\phi}_d)$) aims to mine local label correlations. $\zeta \circ f(\cdot)$ denotes a neural network whose output is then transformed by the softplus function; $\tilde{\boldsymbol{G}}_i$ is the symmetric Laplacian matrix of $\boldsymbol{G}_i$:

$$\begin{aligned} &\boldsymbol{G}_i = \textstyle\sum_{t=1}^{k} \gamma_{it} \mathbf{Y} \mathrm{diag}([\gamma_{1t}; \cdots; \gamma_{nt}]) \mathbf{Y}^\top, \\ &[\gamma_{i1}; \cdots; \gamma_{ik}] = \pi \circ g_i(\mathbf{X}, \mathbf{A}; \boldsymbol{\phi}_{\mathbf{c}}), \end{aligned} \tag{13}$$

where π denotes the softmax function, g is given by a graph convolutional network with adjacency matrix of $\mathbf{A}$ and learnable parameters of $\boldsymbol{\phi}_{\mathbf{c}}$. $\tilde{\boldsymbol{G}}_i \cdot \zeta \circ f(\mathbf{x}_i; \boldsymbol{\phi}_d)$ is essentially a graph convolution operation in which the message passing process works on the labels rather than on the instances. To facilitate understanding Equation (13), we consider the correlation between the i-th and j-th labels for instance $\mathbf{x}_t$ (i.e., the (i, j) entry of $\boldsymbol{G}_t$):

$$\textstyle\sum_{l=1}^{k} \gamma_{tl} \cdot \sum_{s=1}^{n} \gamma_{sl} y_{si} y_{sj}. \tag{14}$$

Equation (14) can be decomposed into the processes of obtaining cluster-specific label correlations and instance-specific label correlations: 1) We first use the linear kernel (i.e., $y_{si} y_{sj}$) to quantify the correlation between the i-th and j-th labels, and then reweight instances by the cluster memberships to obtain the cluster-specific label correlations. 2) We then integrate cluster-specific label correlations by the cluster memberships of a specific instance to obtain the instance-specific label correlation. Note that a potential downside of graph convolutional networks is over-smooth [Li *et al.*, 2019], which can easily derive a label distribution with close description degrees for all labels. Therefore, we additionally incorporate simple labels into the variational posterior, where λ controls the strength of simple labels.

In order to encourage the inference of $\mathbf{c}$ and $\mathbf{D}$ to influence each other through local label correlations, we assume that their variational posteriors share the parameter γ, i.e., $q(\mathbf{c}|\mathbf{X}, \mathbf{A}) = \prod_{i=1}^{n} \mathrm{Cat}(c_i|[\gamma_{i1}; \cdots; \gamma_{ik}])$. Finally, we give the factorization of variational posterior:

$$q(\mathbf{c}, \mathbf{Z}, \mathbf{D}|\mathbf{Y}, \mathbf{X}, \mathbf{A}) = q(\mathbf{Z}|\mathbf{X}, \mathbf{A}) q(\mathbf{D}|\mathbf{X}, \mathbf{Y}) q(\mathbf{c}|\mathbf{X}, \mathbf{A}). \tag{15}$$

According to Equation (10) and Equation (15), we can rewrite the ELBO as:

$$\begin{aligned} \mathcal{L}_{\mathrm{ELBO}} &= \mathbb{E}_{q(\mathbf{Z}|\mathbf{X},\mathbf{A})}[\log p(\mathbf{X}|\mathbf{Z}) + \log p(\mathbf{A}|\mathbf{Z})] \\ &+ \mathbb{E}_{q(\mathbf{D}|\mathbf{X},\mathbf{Y})}[\log p(\mathbf{Y}|\mathbf{D})] \\ &- \mathrm{KL}(q(\mathbf{c}|\mathbf{X}, \mathbf{A}) \| p(\mathbf{c})) - \mathrm{KL}(q(\mathbf{D}|\mathbf{X}, \mathbf{Y}) \| p(\mathbf{D})) \\ &- \mathbb{E}_{q(\mathbf{D}|\mathbf{X},\mathbf{Y}) q(\mathbf{c}|\mathbf{X},\mathbf{A})}[\mathrm{KL}(q(\mathbf{Z}|\mathbf{X}, \mathbf{A}) \| p(\mathbf{Z}|\mathbf{c}, \mathbf{D}))]. \end{aligned} \tag{16}$$

Intuitively, the first term of Equation (16) is the same as in JRC, i.e., Equation (7); The second term of Equation (16) measures the likelihood of reconstructing the simple labels from the label distribution posteriors. This term involves the expectation w.r.t. the Dirichlet posterior $q(\mathbf{D}|\mathbf{X}, \mathbf{Y})$ whose direct reparameterization is difficult unlike the Gaussian posterior [Kingma and Welling, 2014]. Therefore, we decompose the Dirichlet distribution into multiple Gamma distributions which can be reparameterized by their approximated inverse cumulative density function [Joo *et al.*, 2020]:

$$\begin{aligned} &q(\mathbf{D}|\mathbf{X}, \mathbf{Y}) \approx \mathcal{T}_{\mathrm{Dir}}(\mathbf{U}) \text{ whose } (i, j) \text{ entry is} \\ &(u_{ij} \hat{\alpha}_{ij} \Gamma(\hat{\alpha}_{ij}))^{1/\hat{\alpha}_{ij}} \left(\textstyle\sum_{t=1}^{m} (u_{it} \hat{\alpha}_{it} \Gamma(\hat{\alpha}_{it}))^{1/\hat{\alpha}_{it}} \right)^{-1}, \end{aligned} \tag{17}$$

where $u_{ij} \sim \mathrm{Uni}(0, 1)$ is the (i, j) entry of $\mathbf{U}$, $\hat{\alpha}_{ij}$ is the j-th element in $\hat{\boldsymbol{\alpha}}_i$, $\Gamma(\cdot)$ is the Gamma function. Then, we have

$$\mathbb{E}_{q(\mathbf{D}|\mathbf{X},\mathbf{Y})}[\log p(\mathbf{Y}|\mathbf{D})] = L^{-1} \textstyle\sum_{l=1}^{L} \log p(\mathbf{Y}|\mathcal{T}_{\mathrm{Dir}}(\mathbf{U}^{(l)})).$$

Besides, the third and fourth terms of Equation (16) encourage the learned posteriors to incorporate the prior beliefs; the third term has a closed form similar to Equation (8). The forth term is the KL divergence between two Dirichlet distributions, which has a closed form. The fifth term promotes the label distribution to better generate the joint implicit representation, which can be estimated by the reparameterization shown in Equation (17):

$$\begin{aligned} &\mathbb{E}_{q(\mathbf{D}|\mathbf{X},\mathbf{Y}) q(\mathbf{c}|\mathbf{X},\mathbf{A})}[\mathrm{KL}(q(\mathbf{Z}|\mathbf{X}, \mathbf{A}) \| p(\mathbf{Z}|\mathbf{c}, \mathbf{D}))] \\ \approx &L^{-1} \textstyle\sum_{j=1}^{k} \sum_{l=1}^{L} \sum_{i=1}^{n} q(c_i = j|\mathbf{X}, \mathbf{A}) \\ &\mathrm{KL}(q(\mathbf{z}_i|\mathbf{X}, \mathbf{A}) \| p(\mathbf{z}_i|c_i = j, \mathcal{T}_{\mathrm{Dir}}(\mathbf{U}^{(l)}))). \end{aligned} \tag{18}$$

Substituting these equations into Equation (16) yields the optimization objective of LEIC.

4 Experiments

4.1 Experimental Configurations

Datasets and evaluation metrics. Due to page limits, we select six representative real-world LDL datasets from different tasks respectively, and their brief descriptions are shown in Table 1. For Emotion6 and Twitter-LDL, we extract a 168-dimensional feature vector for each instance [Ren *et al.*, 2019]. Besides, we use min-max normalization to preprocess the feature vectors for all datasets to accelerate the convergence. We use four commonly used LDL metrics to measure the similarity or distance between the estimated label distributions and the ground-truth. They are Chebyshev distance (Cheb), KL divergence (KL), cosine coefficient (Cosine), and intersection similarity (Intersec). The first two are distance metrics (i.e., the lower value indicates the better performance), and the last two are similarity metrics (i.e., the higher value indicates the better performance).

Dataset	#Instance	#Feature	#Label
SBU-3DFE [Geng, 2016]	2500	243	6
Emotion6 [Peng *et al.*, 2015]	1980	168	7
Twitter-LDL [Yang *et al.*, 2017]	10045	168	8
Movie [Geng, 2016]	7755	1869	5
Scene [Geng *et al.*, 2022]	2000	294	9
Human Gene [Geng, 2016]	30542	36	68

Table 1: Dataset information.

Comparison algorithms. We compare our algorithm with five LE algorithms, which are GLLE [Xu *et al.*, 2021], LELR [Jia *et al.*, 2023], VLEG [Xu *et al.*, 2023], FLE [Wang *et al.*, 2023], and GLEMR [Lu *et al.*, 2023]. The hyperparameter ranges of GLLE are $\lambda_1 \in \{10^{-4}, 10^{-3}, \cdots, 10^{3}\}$ and $\lambda_2 \in \{10^{-4}, 10^{-3}, \cdots, 10^{3}\}$; The hyperparameters for LELR, VLEG, FLE and GLEMR follow their respective literature. For our JRC and LEIC, k is set to $m+1$, the dimension of the joint implicit representation is set to 64, λ is selected from $\{1, 2, \cdots, 10\}$, neural networks f are modeled as linear functions for simplicity, and Adam [Kingma and Ba, 2015] is adopted as the optimizer.

Experimental method. Recovery and predictive experiments [Xu *et al.*, 2021] are the two basic methods we use to evaluate the performance of LE algorithms. Due to page limits, we only consider the case where the simple label is logical label in the experimental section. In the recovery experiment, we first reduce the ground-truth label distributions in the LDL dataset to logical labels, and then recover label distributions from these logical labels using LE algorithms, and finally compute the distances or similarities between the recovered label distributions and the ground-truth label distributions. In the predictive experiment, we first randomly dividing dataset (70% for training and 30% for testing), and then use LE algorithms to recover the label distributions of training instances which is then used to train an SABFGS [Geng, 2016] model, and finally we repeat the above proess ten times and report the mean predictive performance of SABFGS on test instances.

4.2 Empirical Validation of JRC

Joint Implicit Representations

Here we test the instance representations obtained by JRC.

1) On the one hand, the instance representations obtained by JRC can improve most existing LE algorithms. We run all comparison algorithms based on the original feature vectors and the instance representations obtained by JRC, respectively, and show the recovery performance and predictive performance in Table 3 and Figure 4, respectively. It can be seen that JRC significantly improves the performance of the LE algorithm in most cases. The average improvements in recovery performance ranking are 21.12% (for GLLE), 21.21% (for LELR), 21.01% (for VLEG), 14.01% (for FLE), and 21.95% (for GLEMR), respectively; the average improvements in predictive performance ranking are 15.96% (for GLLE), -1.33% (for LELR), 4.91% (for VLEG), 15.76% (for FLE), and 19.66% (for GLEMR), respectively.

2) On the other hand, the instance representations obtained by JRC can derive better instance relations. In Table 2, we compute the neighborhood divergences of different representations on multiple datasets to measure the quality (w.r.t. the label distribution) of the instance neighborhoods (or instance relations). The neighborhood divergence of a specific instance $\mathbf{x}$ is quantified as the mean of the KL divergences from the neighborhood instances to $\mathbf{x}$ itself w.r.t. the ground-truth label distribution. The "Original features" column and "JRC" column indicate that the instance neighborhoods are determined by the original feature vectors and the instance representations obtained by JRC, respectively. It can be found that JRC considerably reduces the neighborhood divergence; in other words, the JRC-based instance representations can derive better instance relations compared to original features.

	SBU-3DFE	Emotion6	Twitter-LDL	Movie	Scene	Human Gene
JRC	**0.031**	**2.685**	**5.683**	**0.070**	**1.144**	**0.439**
OF	0.130	8.620	9.655	0.218	3.153	0.454

Table 2: Neighborhood divergence formatted as mean±std. The best results are highlighted by boldface. "OF" is the original feature.

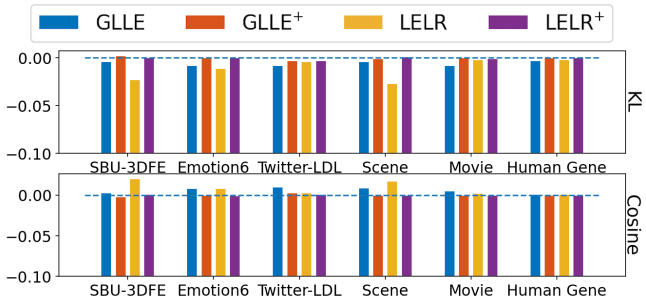


Figure 3: Recovery performance variations arising from JRC-based clustering. Each bar indicates the performance of the corresponding LE algorithm using JRC clustering minus its original performance.

Adaptive Clustering

Here we test the instance clustering obtained by JRC.

1) On the one hand, the instance clustering obtained by JRC can improve most existing LE algorithms that involve local label correlations. Since both GLLE and LELR involve local label correlations, we test the effectiveness of the JRC-based instance clusters for GLLE and LELR. Figure 3 shows the recovery performance variations arising from JRC-based instance clusters. It can be seen that for GLLE and LELR, the performance bars on KL are mostly lower than the horizontal dashed line (which corresponds to 0), and the performance bars on Cosine are mostly higher than the horizontal dashed line, which indicates that JRC-based clustering can improve the recovery performance of GLLE and LELR. Besides, we note that the performance bars of GLLE$^+$ and LELR$^+$ on KL and Cosine are mostly equal to the dashed line, which indicates that JRC-based clustering has almost no effect on GLLE$^+$ and LELR$^+$. This is because GLLE$^+$ and LELR$^+$ are built on top of the JRC-based representations which have clear clustering structures, so that traditional clustering methods also obtain similar results to the JRC-based clustering.

	GLLE	GLLE$^+$	LELR	LELR$^+$	VLEG	VLEG$^+$	FLE	FLE$^+$	GLEMR	GLEMR$^+$	LEIC
SBU-3DFE											
Cheb (↓)	0.119 (10)	0.099 (5)	0.134 (11)	0.108 (8)	0.108 (8)	0.100 (6)	**0.092 (1)**	0.096 (4)	0.103 (7)	**0.092 (1)**	**0.092 (1)**
KL (↓)	0.065 (10)	0.048 (8)	0.085 (11)	0.057 (9)	0.047 (7)	0.043 (5)	**0.039 (1)**	**0.039 (1)**	0.044 (6)	0.042 (4)	**0.039 (1)**
Cosine (↑)	0.936 (10)	0.952 (7)	0.919 (11)	0.943 (9)	0.952 (7)	0.955 (5)	**0.959 (1)**	**0.959 (1)**	0.955 (6)	0.957 (4)	**0.959 (1)**
Intersec (↑)	0.859 (10)	0.877 (8)	0.844 (11)	0.864 (9)	0.881 (7)	0.884 (5)	0.892 (2)	**0.893 (1)**	0.887 (4)	0.884 (5)	0.888 (3)
Emotion6											
Cheb (↓)	0.319 (10)	0.240 (8)	0.317 (9)	0.236 (7)	0.224 (5)	0.223 (4)	0.360 (11)	0.225 (6)	0.188 (3)	0.169 (2)	**0.162 (1)**
KL (↓)	0.599 (10)	0.413 (8)	0.595 (9)	0.403 (6)	0.384 (5)	0.381 (4)	0.995 (11)	0.404 (7)	0.322 (3)	0.292 (2)	**0.285 (1)**
Cosine (↑)	0.722 (10)	0.856 (8)	0.725 (9)	0.863 (6)	0.878 (5)	0.879 (4)	0.601 (11)	0.861 (7)	0.909 (3)	0.919 (2)	**0.932 (1)**
Intersec (↑)	0.566 (10)	0.655 (8)	0.569 (9)	0.661 (7)	0.672 (5)	0.674 (4)	0.558 (11)	0.672 (5)	0.713 (3)	0.749 (2)	**0.751 (1)**
Twitter-LDL											
Cheb (↓)	0.473 (11)	0.415 (8)	0.470 (10)	0.424 (9)	0.406 (7)	0.405 (6)	0.325 (5)	0.281 (4)	0.274 (3)	0.222 (2)	**0.210 (1)**
KL (↓)	1.008 (11)	0.858 (8)	0.998 (10)	0.899 (9)	0.841 (7)	0.838 (6)	0.662 (4)	0.784 (5)	0.558 (3)	0.483 (2)	**0.465 (1)**
Cosine (↑)	0.670 (11)	0.766 (8)	0.676 (10)	0.741 (9)	0.779 (7)	0.781 (6)	0.875 (5)	0.890 (4)	0.904 (3)	0.921 (2)	**0.931 (1)**
Intersec (↑)	0.415 (11)	0.455 (8)	0.418 (10)	0.440 (9)	0.462 (7)	0.463 (6)	0.546 (5)	0.681 (3)	0.612 (4)	0.683 (2)	**0.709 (1)**
Scene											
Cheb (↓)	0.332 (10)	0.316 (9)	0.335 (11)	0.313 (8)	0.310 (6)	0.310 (6)	0.268 (3)	0.308 (5)	0.270 (4)	0.263 (2)	**0.260 (1)**
KL (↓)	0.952 (10)	0.832 (9)	0.963 (11)	0.808 (8)	0.790 (6)	0.788 (5)	0.648 (3)	0.799 (7)	0.668 (4)	0.618 (2)	**0.589 (1)**
Cosine (↑)	0.690 (10)	0.766 (9)	0.686 (11)	0.778 (7)	0.792 (6)	0.793 (5)	0.849 (2)	0.771 (8)	0.828 (4)	0.845 (3)	**0.873 (1)**
Intersec (↑)	0.431 (10)	0.488 (8)	0.429 (11)	0.496 (7)	0.508 (6)	0.509 (5)	0.573 (3)	0.483 (9)	0.555 (4)	0.595 (2)	**0.647 (1)**
Movie											
Cheb (↓)	0.120 (8)	0.113 (7)	0.121 (9)	0.112 (6)	0.099 (4)	0.099 (4)	0.153 (11)	0.128 (10)	0.095 (3)	**0.095 (1)**	**0.095 (1)**
KL (↓)	0.099 (9)	0.089 (7)	0.098 (8)	0.081 (6)	0.068 (4)	0.068 (4)	0.143 (11)	0.104 (10)	0.064 (2)	0.064 (3)	**0.062 (1)**
Cosine (↑)	0.938 (8)	0.947 (7)	0.938 (8)	0.950 (6)	0.963 (4)	0.963 (4)	0.905 (11)	0.936 (10)	0.965 (2)	0.965 (3)	**0.967 (1)**
Intersec (↑)	0.833 (9)	0.844 (7)	0.834 (8)	0.848 (6)	0.864 (4)	0.864 (4)	0.786 (11)	0.823 (10)	0.871 (2)	0.871 (3)	**0.873 (1)**
Human Gene											
Cheb (↓)	0.053 (7)	0.053 (7)	0.053 (7)	0.053 (7)	0.052 (4)	0.051 (2)	0.053 (7)	0.052 (4)	0.051 (3)	0.052 (4)	**0.045 (1)**
KL (↓)	0.236 (9)	0.230 (7)	0.236 (9)	0.232 (8)	0.209 (6)	0.202 (3)	0.236 (9)	0.205 (4)	0.200 (2)	0.205 (4)	**0.150 (1)**
Cosine (↑)	0.835 (9)	0.840 (7)	0.835 (9)	0.838 (8)	0.855 (6)	0.860 (3)	0.835 (9)	0.858 (4)	0.861 (2)	0.858 (4)	**0.898 (1)**
Intersec (↑)	0.786 (9)	0.790 (7)	0.786 (9)	0.789 (8)	0.808 (5)	0.813 (3)	0.786 (9)	0.803 (6)	0.814 (2)	0.811 (4)	**0.818 (1)**

Table 3: The similarity or distance between the ground-truth label distribution and the LE-recovered label distribution. The best performance is highlighted by boldface, and each performance data is followed by its corresponding ranking. In the first row, the LE methods with the superscript "+" indicates that they are based on the instance representations obtained by JRC model.

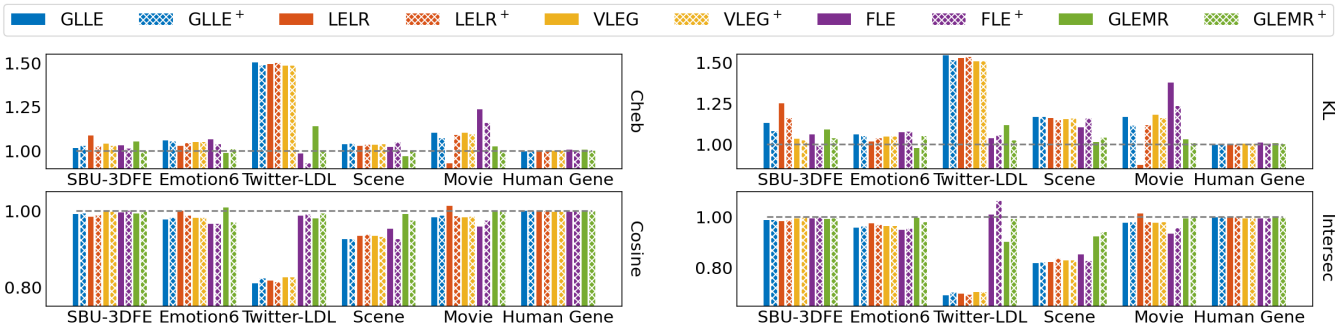


Figure 4: Predictive performance on four LDL measures. The bars with different styles indicate the ratio of the average predictive performance of different LE algorithms relative to LEIC. The gray dashed line indicates the case where the ratio is equal to 1.

2) On the other hand, the instance clustering obtained by JRC has smaller intra-cluster divergence. Table 4 shows the average intra-cluster divergence (i.e., the average of KL divergences from the label distributions of all examples within a specific cluster to their center) of different clustering methods. It can be seen that the average intra-cluster divergence

	SBU-3DFE	Emotion6	Twitter-LDL	Movie	Scene	Human Gene
JRC	**0.038**	**2.859**	**3.958**	**0.105**	**2.221**	**0.150**
KM	0.070	4.681	5.323	0.138	2.551	0.228
GM	0.075	4.929	5.323	0.137	3.038	0.247
HC	0.074	4.649	5.326	0.139	2.416	0.226

Table 4: Average intra-cluster divergence. The best results are highlighted by boldface. KM, GM, and HC denote K-Means, Gaussian mixture clustering, and hierarchical clustering, respectively.

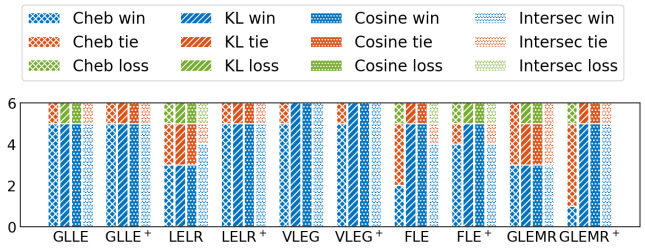


Figure 5: Counts of win/tie/loss for the predictive experiment under pairwise two-tailed t-test with 0.05 significance level.

of JRC-based clustering is significantly smaller than others.

4.3 Empirical Validation of LEIC

Recovery and predictive experiments. Table 3 and Figure 4 show the recovery performance and predictive performance of LEIC, respectively. Figure 5 shows the statistical test results of a pairwise two-tailed t-test with 0.05 significance level. It can be seen that LEIC has competitive advantages both in recovering the label distributions and in serving for the subsequent LDL task.

Parameter analysis and ablation study. To observe the impact of the hyperparameter λ on the recovery performance, we take λ from 0 to 10, and the results are shown in Figure 6. It can be seen that an appropriate addition of λ (e.g., $\lambda \in (0, 4)$) improves the performance on all datasets, which indicates that the simple label information can alleviate the over-smoothing problem of GCN to some extent. To validate the module for mining the local label correlations, we replace Equation (12) with $\hat{\boldsymbol{\alpha}}_i = f([\mathbf{x}_i; \mathbf{y}_i]; \boldsymbol{\phi}_d)$, and report the results in Table 5. It can be seen that using Equation (12) in LEIC has better KL performance and Cosine performance on all datasets than using $\hat{\boldsymbol{\alpha}}_i = f([\mathbf{x}_i; \mathbf{y}_i]; \boldsymbol{\phi}_d)$.

Recovery experiment on the image dataset. To test the performance of our models on non-tabular features, we perform recovery experiments on the original Emotion6 dataset. Specifically, we set the backbone neural networks in JRC and LEIC to ResNet-18 [He *et al.*, 2016], and train comparison algorithms by JRC-based instance representations. The performance is shown in Table 6. Combined with Table 3, it can be seen that the performance on original Emotion6 is better than on the pre-processed tabular data.

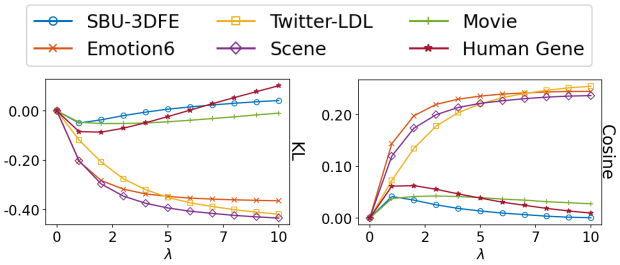


Figure 6: Recovery performance variations caused by changing λ.

Dataset	KL (↓)	Cosine (↑)
SBU-3DFE	**0.039** → 0.050	**0.959** → 0.951
Emotion6	**0.279** → 0.313	**0.935** → 0.929
Twitter-LDL	**0.469** → 0.502	**0.932** → 0.927
Scene	**0.589** → 0.614	**0.873** → 0.870
Movie	**0.062** → 0.073	**0.967** → 0.960
Human Gene	**0.150** → 0.157	**0.898** → 0.894

Table 5: Effectiveness of the local label correlations in LEIC. Each entry is formatted as "$a \rightarrow b$" which denotes that if we replace Equation (12) in LEIC with $\hat{\boldsymbol{\alpha}}_i = f([\mathbf{x}_i; \mathbf{y}_i]; \boldsymbol{\phi}_d)$, the recovery performance will change from a to b.

	GLLE$^+$	LELR$^+$	VLEG$^+$	FLE$^+$	GLEMR$^+$	LEIC
Cheb	0.235	0.233	0.223	0.217	0.167	0.157
KL	0.4	0.396	0.381	0.388	0.284	0.279
Cosine	0.864	0.867	0.879	0.872	0.924	0.935
Intersec	0.662	0.664	0.674	0.671	0.753	0.765

Table 6: Recovery performance on the original Emotion6 dataset.

5 Conclusion

In this paper, we propose a deep generative model called JRC which has following merits: 1) JRC facilitates the acquisition of more accurate instance relations and instance clustering, which can improve most existing LE algorithms; 2) JRC can handdle various forms of features, thus compensating for the shortcomings of the LE algorithms oriented to tabular features. Besides, we also propose a generative LE model called LEIC based on the JRC model. LEIC model mines instance relations and local label correlations by JRC and utilize both relations in a principled way to recover more accurate label distributions. Extensive experiments show that JRC model can improve the performance of most LE algorithms involving instance relations or local label correlations, and LEIC outperforms the state-of-the-art LE algorithms remarkably.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China (62176123, 61906090), the Found of Nanjing University of Aeronautics and Astronautics Research Base Innovation (Science and Technology) Project under Grant NJ2020022, and the Fund of Prospective Layout of Scientific Research for Nanjing University of Aeronautics and Astronautics.

References

[Bengio *et al.*, 2013] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[El Gayar *et al.*, 2006] Neamat El Gayar, Friedhelm Schwenker, and Günther Palm. A study of the robustness of knn classifiers trained using soft labels. In *Artificial Neural Networks in Pattern Recognition*, pages 67–80, 2006.

[Gao *et al.*, 2018] Bin-Bin Gao, Hong-Yu Zhou, Jianxin Wu, and Xin Geng. Age estimation using expectation of label distribution learning. In *International Joint Conference on Artificial Intelligence*, pages 712–718, 2018.

[Geng *et al.*, 2022] Xin Geng, Renyi Zheng, Jiaqi Lv, and Yu Zhang. Multilabel ranking with inconsistent rankers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5211–5224, 2022.

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

[He and Jin, 2019] Tao He and Xiaoming Jin. Image emotion distribution learning with graph convolutional networks. In *International Conference on Multimedia Retrieval*, page 382–390, 2019.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[Hou *et al.*, 2016] Peng Hou, Xin Geng, and Min-Ling Zhang. Multi-label manifold learning. In *AAAI Conference on Artificial Intelligence*, pages 1680–1686, 2016.

[Izenman, 2012] Alan Julian Izenman. Introduction to manifold learning. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(5):439–446, 2012.

[Jia *et al.*, 2019] Xiuyi Jia, Xiang Zheng, Weiwei Li, Changqing Zhang, and Zechao Li. Facial emotion distribution learning by exploiting low-rank label correlations locally. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9841–9850, 2019.

[Jia *et al.*, 2023] Xiuyi Jia, Yunan Lu, and Fangwen Zhang. Label enhancement by maintaining positive and negative label relation. *IEEE Transactions on Knowledge and Data Engineering*, 35(2):1708–1720, 2023.

[Jiang *et al.*, 2006] Xiufeng Jiang, Zhang Yi, and Jian Cheng Lv. Fuzzy svm with a new fuzzy membership function. *Neural Computing & Applications*, 15(3):268–276, 2006.

[Jiang *et al.*, 2017] Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. Variational deep embedding: An unsupervised and generative approach to clustering. In *International Joint Conference on Artificial Intelligence*, pages 1965–1972, 2017.

[Joo *et al.*, 2020] Weonyoung Joo, Wonsung Lee, Sungrae Park, and Il-Chul Moon. Dirichlet variational autoencoder. *Pattern Recognition*, 107:107514, 2020.

[Kingma and Ba, 2015] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[Kingma and Welling, 2014] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

[Li *et al.*, 2019] Guohao Li, Matthias Muller, Ali Thabet, and Bernard Ghanem. Deepgcns: Can gcns go as deep as cnns? In *IEEE/CVF International Conference on Computer Vision*, pages 9267–9276, 2019.

[Liu *et al.*, 2021a] Xinyuan Liu, Jihua Zhu, Zhongyu Li, Zhiqiang Tian, Xiuyi Jia, and Lei Chen. Unified framework for learning with label distribution. *Information Fusion*, 75:116–130, 2021.

[Liu *et al.*, 2021b] Xinyuan Liu, Jihua Zhu, Qinghai Zheng, Zhongyu Li, Ruixin Liu, and Jun Wang. Bidirectional loss function for label enhancement and distribution learning. *Knowledge-Based System*, 213:106690, 2021.

[Lu and Jia, 2022] Yunan Lu and Xiuyi Jia. Predicting label distribution from multi-label ranking. In *Advances in Neural Information Processing Systems*, 2022.

[Lu *et al.*, 2023] Yunan Lu, Liang He, Fan Min, Weiwei Li, and Xiuyi Jia. Generative label enhancement with gaussian mixture and partial ranking. In *AAAI Conference on Artificial Intelligence*, 2023.

[Luo *et al.*, 2021] Jianqiao Luo, Yihan Wang, Yang Ou, Biao He, and Bailin Li. Neighbor-based label distribution learning to model label ambiguity for aerial scene classification. *Remote Sensing*, 13(4):755, 2021.

[Lv *et al.*, 2019] Jiaqi Lv, Ning Xu, Renyi Zheng, and Xin Geng. Weakly supervised multi-label learning via label enhancement. In *International Joint Conference on Artificial Intelligence*, pages 3101–3107, 2019.

[Peng *et al.*, 2015] Kuan-Chuan Peng, Tsuhan Chen, Amir Sadovnik, and Andrew Gallagher. A mixed bag of emotions: Model, predict, and transfer emotion distributions. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 860–868, 2015.

[Plackett, 1975] Robin L. Plackett. The analysis of permutations. *Applied Statistics*, 24(2):193–202, 1975.

[Ren *et al.*, 2019] Tingting Ren, Xiuyi Jia, Weiwei Li, Lei Chen, and Zechao Li. Label distribution learning with label-specific features. In *International Joint Conference on Artificial Intelligence*, pages 3318–3324, 2019.

[Shao *et al.*, 2018] Ruifeng Shao, Xin Geng, and Ning Xu. Multi-label learning with label enhancement. In *IEEE International Conference on Data Mining*, pages 437–446, 2018.

[Shen *et al.*, 2021] Wei Shen, Yiluan Guo, Yan Wang, Kai Zhao, Bo Wang, and Alan Yuille. Deep differentiable random forests for age estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(2):404–419, 2021.

[Tang *et al.*, 2020] Haoyu Tang, Jihua Zhu, Qinghai Zheng, Jun Wang, Shanmin Pang, and Zhongyu Li. Label enhancement with sample correlations via low-rank representation. In *AAAI Conference on Artificial Intelligence*, pages 5932–5939, 2020.

[Wang *et al.*, 2023] Ke Wang, Ning Xu, Miaogen Ling, and Xin Geng. Fast label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 35(2):1502–1514, 2023.

[Wen *et al.*, 2020] Xin Wen, Biying Li, Haiyun Guo, Zhiwei Liu, Guosheng Hu, Ming Tang, and Jinqiao Wang. Adaptive variance based label distribution learning for facial age estimation. In *European Conference on Computer Vision*, page 379–395, 2020.

[Wen *et al.*, 2021] Tao Wen, Weiwei Li, Lei Chen, and Xiuyi Jia. Semi-supervised label enhancement via structured semantic extraction. *International Journal of Machine Learning and Cybernetics*, 13:1131–1144, 2021.

[Xu *et al.*, 2018] Ning Xu, An Tao, and Xin Geng. Label enhancement for label distribution learning. In *International Joint Conference on Artificial Intelligence*, pages 1632–1643, 2018.

[Xu *et al.*, 2019] Ning Xu, Jiaqi Lv, and Xin Geng. Partial label learning via label enhancement. In *AAAI Conference on Artificial Intelligence*, pages 5557–5564, 2019.

[Xu *et al.*, 2020] Ning Xu, Jun Shu, Yun-Peng Liu, and Xin Geng. Variational label enhancement. In *International Conference on Machine Learning*, volume 119, pages 10597–10606, 2020.

[Xu *et al.*, 2021] Ning Xu, Yun-Peng Liu, and Xin Geng. Label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 33:1632–1643, 2021.

[Xu *et al.*, 2023] Ning Xu, Jun Shu, Renyi Zheng, Xin Geng, Deyu Meng, and Min-Ling Zhang. Variational label enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(5):6537–6551, 2023.

[Yang *et al.*, 2017] Jufeng Yang, Ming Sun, and Xiaoxiao Sun. Learning visual sentiment distributions via augmented conditional probability neural network. In *AAAI Conference on Artificial Intelligence*, pages 224–230, 2017.

[Zhang *et al.*, 2018] Qian-Wen Zhang, Yun Zhong, and Min-Ling Zhang. Feature-induced labeling information enrichment for multi-label learning. In *AAAI Conference on Artificial Intelligence*, pages 4446–4453, 2018.

[Zhang *et al.*, 2021] Min-Ling Zhang, Qian-Wen Zhang, Jun-Peng Fang, Yukun Li, and Xin Geng. Leveraging implicit relative labeling-importance information for effective multi-label learning. *IEEE Transactions on Knowledge and Data Engineering*, 33:2057–2070, 2021.

[Zheng *et al.*, 2023] Qinghai Zheng, Jihua Zhu, Haoyu Tang, Xinyuan Liu, Zhongyu Li, and Huimin Lu. Generalized label enhancement with sample correlations. *IEEE Transactions on Knowledge and Data Engineering*, 35(1):482–495, 2023.