# Collaborative Learning of Label Semantics and Deep Label-Specific Features for Multi-Label Classification

Jun-Yi Hang, and Min-Ling Zhang, *Senior Member, IEEE*

**Abstract**—In multi-label classification, the strategy of *label-specific features* has been shown to be effective to learn from multi-label examples by accounting for the distinct discriminative properties of each class label. However, most existing approaches exploit the semantic relations among labels as immutable prior knowledge, which may not be appropriate to constrain the learning process of label-specific features. In this paper, we propose to learn label semantics and label-specific features in a collaborative way. Accordingly, a deep neural network (DNN) based approach named CLIF, i.e. ***C**ollaborative **L**earning of label semant**I**cs and deep label-specific **F**eatures for multi-label classification*, is proposed. By integrating a graph autoencoder for encoding semantic relations in the label space and a tailored feature-disentangling module for extracting label-specific features, CLIF is able to employ the learned label semantics to guide mining label-specific features and propagate label-specific discriminative properties to the learning process of the label semantics. In such a way, the learning of label semantics and label-specific features interact and facilitate with each other so that label semantics can provide more accurate guidance to label-specific feature learning. Comprehensive experiments on 14 benchmark data sets show that our approach outperforms other well-established multi-label classification algorithms.

**Index Terms**—Machine learning, multi-label classification, label-specific features, label semantics, collaborative learning.

## 1 Introduction

MULTI-LABEL classification aims to derive classification models from instances associated with multiple class labels simultaneously [1]. As a practical machine learning paradigm, multi-label classification has been widely applied in various real-world applications, such as multimedia content annotation [2], [3] where the task is to recognize all objects occurring in an image, text categorization [4], [5] where each document may cover several topics, music emotion analysis [6], [7] where a song may express various emotions, etc.

One common strategy to learn from multi-label data is to employ the identical feature set of the instance to induce classification models. Although feasible results have been achieved in such multi-label classification approaches, this strategy might be suboptimal as it fails to account for the distinct characteristics of each class label. For example, in automatic image annotation, *shape*-based features would be more essential in recognizing the *plane* category, while *color*-based features might be preferred in discriminating the *sky* category. With the ability to model distinct discriminative properties of each class label, label-specific feature learning, which aims to find the most pertinent and discriminative features specific to each class label, has become a promising strategy to facilitate multi-label classification [8], [9], [10].

Some early approaches construct label-specific features heuristically. For example, LIFT [8] firstly performs clustering analysis on positive and negative instances of each label, and then obtains label-specific features via querying distances between the original instances and the cluster centers. To improve this, many approaches have been developed to learn label-specific features by exploring the semantic relations among labels, where the label correlations are exploited as prior knowledge to constrain the learning process of label-specific features [11], [12], [13], [14], [15]. More specifically, these approaches calculate the similarity between pairwise labels and incorporate these similarity-based label correlations into model training, where constraints are imposed to share more features [9], [16], [17] or similar predictions [10] among strongly correlated labels. Nevertheless, these existing approaches merely introduce the semantic relations among labels via a precomputed similarity matrix in label space, which may not be appropriate for downstream task, i.e. label-specific feature learning.

To address above issues, we propose to collaboratively learn label semantics and label-specific features. Concretely, the label semantics derived from label space are employed to actively guide finding the most discriminative features for each class label, while the discrimination process based on these label-specific features in turn affects the learning process of the label semantics so that label semantics can provide more accurate guidance to label-specific feature learning.

Following this strategy, a DNN-based approach named CLIF, i.e. ***C**ollaborative **L**earning of label semant**I**cs and deep label-specific **F**eatures for multi-label classification*, is presented. In CLIF, we introduce a graph autoencoder to encode the rich semantic dependencies among labels into semantic label embeddings which capture correlations in label space and we develop a tailored feature-disentangling module to

- *Jun-Yi Hang and Min-Ling Zhang are with the School of Computer Science and Engineering, Southeast University, Nanjing 210096, China, and the Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China. (Corresponding author: Min-Ling Zhang)*
  *Email: {hangjy, zhangml}@seu.edu.cn*

*Manuscript received.*

extract label-specific features. In a collaborative way, learned label embeddings are employed to guide the selection of the most pertinent features of each class label and label-specific discriminative properties are incorporated into the correlation-aware label embeddings via backpropagation of discrimination errors.

In this paper, we advance the label-specific feature learning to the deep learning scenario, which is a promising area that has not received much attention. Relying on the end-to-end learnable properties of deep neural networks, CLIF makes a first attempt towards the appealing collaborative learning strategy. Comprehensive experiments on 14 benchmark data sets show that CLIF performs better than well-established multi-label classification algorithms.

The rest of this paper is organized as follows. Section 2 briefly reviews related works. Section 3 presents details of the proposed CLIF approach. Section 4 reports experimental results over a wide range of multi-label data sets. Section 5 concludes this paper.

## 2 RELATED WORKS

As a practical and challenging machine learning paradigm, multi-label classification has been studied extensively in recent years [1]. To cope with an output space which is exponential in size to the number of class labels, numerous approaches propose exploiting label correlations to improve the learning process [18], [19], [20], [21], [22]. Generally speaking, the order of label correlations considered in existing approaches can be grouped into three categories, namely *first-order* correlations [23], [24], *second-order* correlations [25], [26], [27] and *high-order* correlations [28], [29], [30].

Complementary to label correlation exploitation, manipulating the feature space is also an attractive way to facilitate multi-label classification. Dimensionality reduction [31] or feature selection [32] approaches focus on learning more compact representations for the original features. While some embedding-based approaches [33], [34], [35] map the original features into a semantic space where semantically similar instances are close to each other. Other approaches exist that introduce multi-view representations [36], [37] or learn meta-level features [38], [39] for multi-label data.

As an alternative strategy for feature manipulation, label-specific features differentiate itself from the above feature manipulation strategies via tailoring features specific to each class label. The basic assumption behind label-specific features is that each class label may possess distinct discriminative properties. Therefore, a more effective multi-label classification model can be induced if the most pertinent and discriminative features for each class label could be provided. As the seminal work, LIFT [8] heuristically constructs the label-specific features by querying cluster centers of each class label. Successively, several approaches have been proposed to improve the construction process of label-specific features. To alleviate the increasing of feature dimensionality encountered in LIFT, the fuzzy rough set is introduced to perform label-specific feature reduction [40]. Some works aim to optimize the unstable clustering process of *k*-means via clustering ensemble [41], [42]. In addition, some other works augment label-specific features with local neighbor information [43], global spatial topology information [44], or informative features from related class labels [14].

Another line of research formalizes label-specific feature construction as label-specific feature selection, i.e. retaining a specific subset of the original features for each class label [11], [12], [15], [45]. LLSF [9] presents a framework based on lasso regression for label-specific feature selection with feature-sharing between closely-related labels. JFSC [16] further incorporates extra Fisher discriminant-based regularization term into the feature selection process. Furthermore, there have been other strategies for enhancing label-specific feature selection such as introducing spectral clustering for feature selection over meta-labels [46], imposing non-sparse constraints on the selected feature subsets [47], or directly regularizing the predictions with label correlations [10], etc. It is worth noting that existing approaches merely exploit the semantic relations of labels as prior knowledge to impose constraints on the learning process of label specific features. In other words, these label semantics are immutable during the whole learning process.

Furthermore, the strategy of label-specific features has also been jointly considered with the extreme multi-label learning problem, where the label space may possess over millions of labels. To optimize the excessive algorithmic complexity brought by constructing label-specific features, [48] reorganizes the label space into a probabilistic label tree and captures the most relevant part of text for each meta-label in the tree via a multi-label attention mechanism. While [49] relies on label-specific features to enhance the learning process of tail labels, which are ubiquitous in extreme multi-label data sets.

Recently, deep learning has become a successful technique to solve the multi-label classification problem [17], [50], [51], [52], [53], [54], [55]. Dates back to [56], deep neural networks have been competent to construct a latent embedding space which can well capture the dependency between the features and labels [57], [58]. Success has been witnessed in deep embedding-based approaches, such as C2AE [59] and MPVAE [60], which resort to deep neural networks to learn and align the latent spaces for features and labels. Several works focus on exploiting deep neural networks to capture label correlations [61], [62], [63]. For instance, ML-GCN [64] introduces graph neural networks to explicitly encode the label correlations into inter-dependent classifiers. Sequential prediction methods such as [65], [66] utilize recurrent neural networks to better exploit the higher-order label dependencies.

Due to the powerful representation learning capability of deep neural networks, it is quite natural to consider the problem of label-specific features in the deep learning scenario. However, this is still an area that has not received much attention. In the next section, a first attempt towards deep label-specific feature learning with collaborative learning strategy will be introduced in detail.

## 3 THE CLIF APPROACH

### 3.1 Notations

The following notations are used in the rest of this paper. Let $\mathcal{X} = \mathbb{R}^d$ denote the input space and $\mathcal{Y} = \{l_1, l_2, \ldots, l_q\}$ denote the label space with $q$ class labels. A multi-label

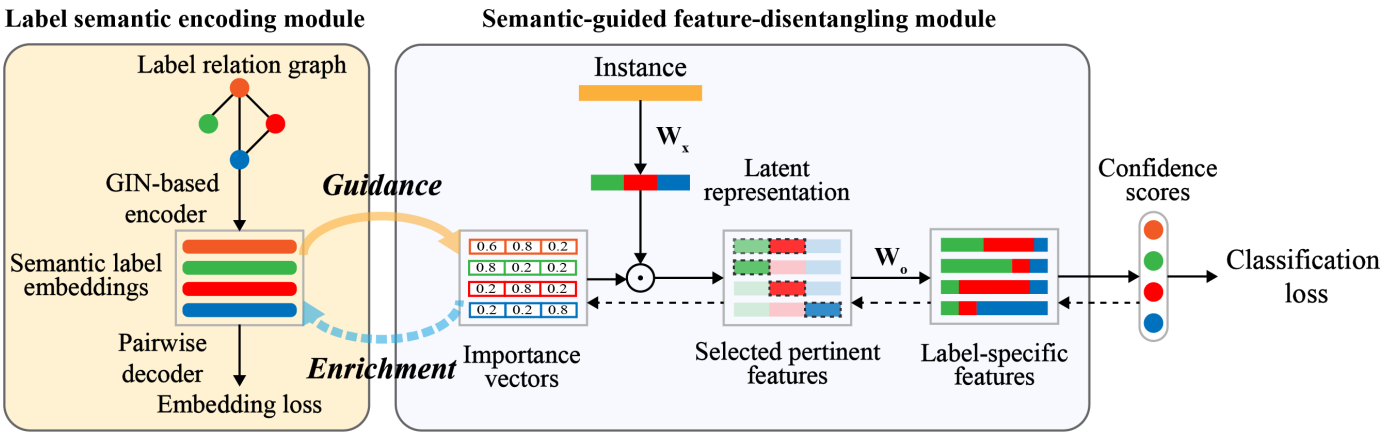


Fig. 1. Illustration of the proposed CLIF approach. CLIF learns label semantics and label-specific features collaboratively. On one hand, the semantic-guided feature-disentangling module extracts label-specific features with the guidance of semantic label embeddings generated by the label semantic encoding module. On the other hand, as shown by the dotted arrows, discrimination process based on label-specific features in turn enriches the label semantics with label-specific discriminative properties via backpropagation of discrimination errors.

example is denoted as $(\mathbf{x}, Y)$, where $\mathbf{x} \in \mathcal{X}$ is its feature vector and $Y \subseteq \mathcal{Y}$ is its set of relevant labels. Here, a $q$-dimensional vector $\mathbf{y} = [y_1, y_2, \ldots, y_q] \in \{0,1\}^q$ is utilized to denote $Y$, where $y_k = 1$ indicates $l_k \in Y$ and $y_k = 0$ otherwise. Formally, multi-label classification aims to derive a multi-label prediction function $h : \mathcal{X} \to 2^{\mathcal{Y}}$ from a multi-label data set $\mathcal{D} = \{(\mathbf{x}_i, Y_i) | 1 \leq i \leq m\}$. Given an unseen instance $\mathbf{u} \in \mathcal{X}$, its associated label set is predicted as $h(\mathbf{u}) \subseteq \mathcal{Y}$.

### 3.2 Overview

Fig. 1 gives an overall illustration of the proposed CLIF approach. In the label semantic encoding module, a graph based on label co-occurrence is constructed over the label space and then is fed into a graph autoencoder to generate semantic-related label embeddings. With the guidance of semantic label embeddings, the representation of an instance is disentangled into label-semantic aware label-specific features in the semantic-guided feature-disentangling module. By training the whole model end-to-end, discrimination errors on label-specific features are backpropagated into the label semantic encoding module so that semantic label embeddings can be updated to capture each label's own discriminative properties accurately. In such a collaborative way, pertinent features for each class label can be mined thoroughly with simultaneously improved label semantics. We will describe key modules in CLIF in detail.

### 3.3 Label Semantic Encoding

Label co-occurrence is an essential semantic relation in label space, modeling of which lies in the heart of multi-label classification. In this section, we attempt to learn the label embeddings in a label co-occurrence semantic space, where labels with strong co-occurrence relationship possess similar embeddings.

To achieve this, we firstly construct a label relation graph based on statistics of label co-occurrence. Let $G = (V, E)$ denote such a label relation graph, where $V$ denotes the set of nodes corresponding to the set of class labels and $E$ denotes the set of edges. The adjacency matrix $\mathbf{A}$ stores the weights associated with each edge, representing the strengths of co-occurrence relationship between pairs of labels. In this paper, we formulate the adjacency matrix $\mathbf{A}$ as the symmetric conditional probability matrix[1]

$$\mathbf{A}_{ij} = \frac{1}{2}[P(l_j|l_i) + P(l_i|l_j)]$$

where $P(l_j|l_i)$ is the probability that label $l_j$ appears when label $l_i$ appears and the diagonal elements of conditional probability matrix $\mathbf{P}$ are set to 0. We calculate the conditional probability matrix $\mathbf{P}$ on training set.

Successively, a graph autoencoder is applied to embed labels into a label co-occurrence semantic space with the label relation graph. The encoder in our graph autoencoder is instantiated by Graph Isomorphism Network (GIN) [67], which was originally designed for graph classification task with the most powerful representational capacity provably. We introduce GIN to capture the label correlations contained in the label relation graph.

Given a feature matrix of nodes $\mathbf{H}^{(t)} \in \mathbb{R}^{q \times d^{(t)}}$ where each row corresponds to the embedding of a label and $d^{(t)}$ denotes the dimensionality of node features, together with the adjacency matrix $\mathbf{A}$, a GIN layer updates node features by

$$\mathbf{H}^{(t+1)} = f^{(t+1)}[(1 + \epsilon^{(t+1)})\mathbf{H}^{(t)} + \mathbf{A}\mathbf{H}^{(t)}] \quad (1)$$

where $\mathbf{H}^{(t+1)} \in \mathbb{R}^{q \times d^{(t+1)}}$ is the updated feature matrix of nodes, $f^{(t+1)}$ denotes a neural network consisting of two fully-connected layers followed by Batch Normalization [68] and LeakyReLU activation [69], and $\epsilon^{(t+1)}$ is a learnable parameter which controls the importance of node's own features during neighborhood aggregation. The initial feature matrix of nodes $\mathbf{H}^{(0)} \in \mathbb{R}^{q \times d^{(0)}}$ is initialized by Gaussian function with zero mean and standard deviation of 1, which has better empirical performance than one-hot embeddings. After stacking GIN layers, we take $\mathbf{H}^{(T)} \in \mathbb{R}^{q \times d^{(T)}}$ as the

1. Actually, the adjacency matrix $\mathbf{A}$ can be constructed in numerous alternative ways or even can be implemented in a learnable formulation. We attempt to focus on the collaborative learning process for label semantics and label-specific features and will leave it for further work.

final label embeddings $\mathbf{E} \in \mathbb{R}^{q \times d_e}$, i.e. $\mathbf{E} = \mathbf{H}^{(T)}$, for downstream feature-disentangling process.

By sharing layer parameters among all the class labels and explicitly incorporating label correlations into the adjacency matrix, GIN is able to embed labels with strong co-occurrence relationship to nearby locations in the label semantic space. However, the embeddings of weakly correlated labels cannot be effectively pushed away from each other, as the neighborhood aggregation scheme of GIN does not ensure increased distinction between two non-adjacent labels. Therefore, a pairwise decoder is utilized to ensure that the learned semantic label embeddings capture the topological structure of the semantic space well. The objective function of the pairwise decoder is formulated as follows

$$\mathcal{L}_{le} = \frac{1}{q^2} \sum_{i=1}^{q} \sum_{j=1}^{q} [cos(\mathbf{e}_i, \mathbf{e}_j) - \hat{\mathbf{A}}_{ij}]^2 \tag{2}$$

where $cos(\mathbf{e}_i, \mathbf{e}_j)$ denotes the cosine similarity between the label embeddings of label $l_i$ and $l_j$, and $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ with $\mathbf{I}$ being an identity matrix.

### 3.4 Semantic-Guided Feature-Disentangling

In the semantic-guided feature-disentangling module, label semantics provide guidance and incorporate label correlations to the learning process of label-specific features. Meanwhile, this module serves as a bridge to propagate discrimination errors on label-specific features to the label semantic encoding module.

Here, for each class label $l_k \in \mathcal{Y}$, a specific mapping $\phi_k : \mathcal{X} \to \mathcal{Z}_k$ from the original feature space $\mathcal{X} \in \mathbb{R}^d$ to the label-specific feature space $\mathcal{Z}_k \in \mathbb{R}^{d_z}$ is learned. With the guidance of the semantic label embeddings $\mathbf{E}$ generated by the label semantic encoding module, label-specific mapping $\phi_k$ can be formulated as

$$\phi_k(\mathbf{x}) = \phi(\mathbf{x}, \mathbf{e}_k)$$

where $\mathbf{e}_k$ denotes the label embedding of label $l_k$ which corresponds to the $k^{th}$ row of $\mathbf{E}$.

To achieve this, We firstly map instance representation from the original feature space to a more powerful deep latent space, where the latent representation can be optimized to benefit the subsequent element-wise selection process of the pertinent features for each class label

$$\mathbf{x}_z = \zeta(\mathbf{W}_x \mathbf{x} + \mathbf{b}_x)$$

where $\mathbf{W}_x \in \mathbb{R}^{d_z \times d}$, $\mathbf{b}_x \in \mathbb{R}^{d_z}$ are learnable parameters shared among all the class labels, and $\zeta$ is the LeakyReLU activation. Then, an attention-like mechanism is designed to utilize label semantics to guide selecting the pertinent and discriminative features for each class label. Concretely, we exploit a one-layer fully-connected network to decode the semantic label embeddings into feature importance vectors

$$\alpha_k = \sigma(\mathbf{W}_e \mathbf{e}_k + \mathbf{b}_e)$$

where $\alpha_k \in \mathbb{R}^{d_z}$ is the feature importance vectors specific to label $l_k$, $\mathbf{W}_e \in \mathbb{R}^{d_z \times d_e}$, $\mathbf{b}_e \in \mathbb{R}^{d_z}$ are shared learnable parameters, and $\sigma$ denotes the sigmoid function which can generate multiple peaks to keep consistency with the fact that there might be multiple pertinent features for each class label. Successively, pertinent features for each class label are selected via Hadamard product between the feature importance vectors and instance latent representation. Finally, we feed pertinent features into another one-layer fully-connected network to obtain the final label-specific features

$$\mathbf{z}_k = \zeta[\mathbf{W}_o(\mathbf{x}_z \odot \alpha_k) + \mathbf{b}_o]$$

where $\mathbf{W}_o \in \mathbb{R}^{d_z \times d_z}$, $\mathbf{b}_o \in \mathbb{R}^{d_z}$ are shared learnable parameters, and $\odot$ denotes the Hadamard product.

### 3.5 Classification

For each class label $l_k \in \mathcal{Y}$, a fully-connected layer is attached to the corresponding label-specific features $\mathbf{z}_k$ to predict the confidence score of the presence of label $l_k$, formulated as

$$s_k = g_k(\mathbf{z}_k) = \sigma(\mathbf{w}_k^{\mathrm{T}} \mathbf{z}_k + b_k)$$

where $\mathbf{w}_k \in \mathbb{R}^{d_z}$ and $b_k$ are learnable parameters, $\sigma$ denotes the sigmoid function. Given an unseen instance $\mathbf{u} \in \mathcal{X}$, its associated label set is predicted as

$$Y = \{l_k | g_k[\phi_k(\mathbf{u})] > 0.5, 1 \leq k \leq q\}$$

### 3.6 Overall Objective Function

CLIF is trained with the following objective function in an end-to-end fashion

$$\mathcal{L} = \mathcal{L}_{ce} + \lambda \mathcal{L}_{le}$$

where $\mathcal{L}_{le}$ is the label embedding loss elaborated in Eq. (2), $\lambda$ is a trade-off parameter, and $\mathcal{L}_{ce}$ denotes the cross entropy loss, formulated as

$$\mathcal{L}_{ce} = -\sum_{k=1}^{q} y_k \log s_k + (1 - y_k) \log(1 - s_k) \tag{3}$$

## 4 EXPERIMENTS

### 4.1 Experimental Configurations

#### 4.1.1 Data Sets

Table 1 summarizes detailed characteristics of the 14 multi-label data sets used in the experiments. Properties of each data set are characterized by several statistics, including number of examples $|\mathcal{S}|$, number of features $dim(\mathcal{S})$, number of possible class labels $L(\mathcal{S})$, feature type $F(\mathcal{S})$, label cardinality (average number of labels per instance) $LCard(\mathcal{S})$, label density (label cardinality over $L(\mathcal{S})$) $LDen(\mathcal{S})$, number of distinct label sets $DL(\mathcal{S})$ and proportion of distinct label sets $PDL(\mathcal{S})$. Detailed definitions on these statistics can be found in [1].

Following [8], we perform dimensionality reduction for rcv-s1 and tmc2007 by retaining the top 2% features with highest document frequency. For iaprtc12, espgame and mirflickr, the local descriptor *DenseSift* is used. As shown in Table 1, the 14 multi-label data sets possess diversified multi-label properties. Therefore, experimental studies on these data sets provide a solid basis for comprehensive evaluation of CLIF's effectiveness.

TABLE 1
Characteristics of the Experimental Data Sets

| Dataset | $\|\mathcal{S}\|$ | $dim(\mathcal{S})$ | $L(\mathcal{S})$ | $F(\mathcal{S})$ | $LCard(\mathcal{S})$ | $LDen(\mathcal{S})$ | $DL(\mathcal{S})$ | $PDL(\mathcal{S})$ | Domain |
|---|---|---|---|---|---|---|---|---|---|
| CAL500 | 502 | 68 | 174 | Numeric | 26.044 | 0.150 | 502 | 1.000 | Music[1] |
| Image | 2000 | 294 | 5 | Numeric | 1.236 | 0.247 | 20 | 0.010 | Images[2] |
| scene | 2407 | 294 | 6 | Numeric | 1.074 | 0.179 | 15 | 0.006 | Images[1] |
| yeast | 2417 | 103 | 14 | Numeric | 4.237 | 0.303 | 198 | 0.082 | Biology[1] |
| corel5k | 5000 | 499 | 374 | Nominal | 3.522 | 0.009 | 3175 | 0.635 | Images[1] |
| rcv1-s1 | 6000 | 944 | 101 | Numeric | 2.880 | 0.029 | 1028 | 0.171 | Text[1] |
| Corel16k-s1 | 13766 | 500 | 153 | Nominal | 2.859 | 0.019 | 4803 | 0.349 | Images[1] |
| delicious | 16105 | 500 | 983 | Nominal | 19.020 | 0.019 | 15806 | 0.981 | Text[1] |
| iaprtc12 | 19627 | 1000 | 291 | Numeric | 5.719 | 0.020 | 16202 | 0.825 | Images[3] |
| espgame | 20770 | 1000 | 268 | Numeric | 4.686 | 0.017 | 18158 | 0.874 | Images[3] |
| mirflickr | 25000 | 1000 | 38 | Numeric | 4.716 | 0.124 | 4464 | 0.179 | Images[3] |
| tmc2007 | 28596 | 981 | 22 | Nominal | 2.158 | 0.098 | 1341 | 0.047 | Text[1] |
| mediamill | 43907 | 120 | 101 | Numeric | 4.376 | 0.043 | 6555 | 0.149 | Video[1] |
| bookmarks | 87856 | 2150 | 208 | Nominal | 2.028 | 0.010 | 18716 | 0.213 | Text[1] |

[1] http://mulan.sourceforge.net/datasets.html
[2] http://palm.seu.edu.cn/zhangml/
[3] http://lear.inrialpes.fr/people/guillaumin/data.php

#### 4.1.2 Evaluation Metrics

Six widely-used evaluation metrics for multi-label classification are employed to evaluate the performance of each approach, including *One-error*, *Coverage*, *Ranking loss*, *Average precision*, *Macro-averaging AUC* and *Adjusted hamming loss*[2]. Given the test data set $\mathcal{T} = \{(\mathbf{x}_i, Y_i)|1 \leq i \leq n\}$ and the learned prediction functions $\{f_1, f_2, \ldots, f_q\}$ for each class label respectively, these evaluation metrics are formulated as follows:

- *One-error*: $\frac{1}{n}\sum_{i=1}^{n} \mathbb{I}[(\arg\max_{l_k \in \mathcal{Y}} f_k(\mathbf{x}_i)) \notin Y_i]$ where $\mathbb{I}(\mathcal{P}) = 1$ if predicate $\mathcal{P}$ holds and $\mathbb{I}(\mathcal{P}) = 0$ otherwise.
- *Coverage*: $\frac{1}{q}[\frac{1}{n}\sum_{i=1}^{n} \max_{l_k \in Y_i} rank(\mathbf{x}_i, l_k) - 1]$ where $rank(\mathbf{x}_i, l_k) = \sum_{j=1}^{q} \mathbb{I}[f_j(\mathbf{x}_i) \geq f_k(\mathbf{x}_i)]$.
- *Ranking loss*: $\frac{1}{n}\sum_{i=1}^{n} \frac{|\mathcal{Z}_i|}{|Y_i||\bar{Y}_i|}$ where $\bar{Y}_i = \mathcal{Y} \backslash Y_i$ and $\mathcal{Z}_i = \{(l_k, l_j)|f_k(\mathbf{x}_i) \leq f_j(\mathbf{x}_i), (l_k, l_j) \in Y_i \times \bar{Y}_i\}$.
- *Average precision*: $\frac{1}{n}\sum_{i=1}^{n} \frac{1}{|Y_i|}\sum_{l_k \in Y_i} \frac{|\mathcal{R}(\mathbf{x}_i, l_k)|}{rank(\mathbf{x}_i, l_k)}$ where $\mathcal{R}(\mathbf{x}_i, l_k) = \{l_j|f_j(\mathbf{x}_i) \geq f_k(\mathbf{x}_i), l_j \in Y_i\}$.
- *Macro-averaging AUC*: $\frac{1}{q}\sum_{k=1}^{q} auc_k$ where $auc_k = \frac{|\{(\mathbf{x}', \mathbf{x}'')|f_k(\mathbf{x}') \geq f_k(\mathbf{x}'')\}|}{|\mathcal{P}_k||\mathcal{N}_k|}$, $\mathcal{P}_k$ and $\mathcal{N}_k$ consist of the instances with and without label $l_k$ respectively, $(\mathbf{x}', \mathbf{x}'') \in \mathcal{P}_k \times \mathcal{N}_k$.
- *Adjusted hamming loss*: $\frac{1}{n}\sum_{i=1}^{n} \frac{|\hat{Y}_i \Delta Y_i|}{|\hat{Y}_i| + |Y_i|}$ where $\hat{Y}_i$ denotes the predicted set of relevant labels for $\mathbf{x}_i$ and $\Delta$ denotes the symmetric difference between two sets.

All the above multi-label metrics take values in $[0, 1]$. For Average precision and Macro-averaging AUC, *larger* values mean better performance. While for the other four metrics, *smaller* values indicate better performance.

#### 4.1.3 Implementation Details

Unless otherwise stated, we stack two GIN layers to encode semantic label embeddings and residual connection is added between these GIN layers. The dimensionality of the initial node features is set to be equal to the number of class labels. i.e. $d^{(0)} = q$, while the output dimensionalities of the two-layer neural networks $\{f^{(1)}, f^{(2)}\}$ in GIN layers are both set as $d_e$. The dimensionality of the label-specific features is set as 512. All the LeakyReLU activation functions have a negative slope of 0.1. We initialize all the learnable layers with the MSRA method [71] and initialize the learnable parameters $\{\epsilon^{(1)}, \epsilon^{(2)}\}$ in GIN by 0. For network optimization, Adam with a batch size of 1000, momentums of 0.999 and 0.9 is employed. The learning rate is set as $10^{-3}$ and the weight decay is set as $10^{-5}$.

### 4.2 Comparative Studies

We compare CLIF[3] against six state-of-the-art multi-label classification approaches with parameter configurations suggested in respective literatures:

- LIFT [8]: LIFT constructs label-specific features via querying clustering results on the positive and negative instances of each label. [parameter configuration: $r = 0.1$]
- LLSF [9], [15]: LLSF performs label-specific feature selection in a lasso-regression-like framework with feature-sharing between closely-related labels. [parameter configuration: $\alpha = 0.1$, $\beta = 0.1$, $\gamma = 0.01$]
- JFSC [16]: JFSC performs label-specific feature selection and classification jointly with pairwise label correlations. [parameter configuration: grid search for $\alpha, \beta, \gamma \in \{4^{-5}, 4^{-4}, \ldots, 4^{5}\}$ and $\eta \in \{0.1, 1, 10\}$]

2. As an enhanced version of the conventional Hamming loss, Adjusted hamming loss [70] is more sensitive to performance differences among approaches when the data set has low label density.

3. Code of CLIF is publicly available at: http://palm.seu.edu.cn/zhangml/files/CLIF.rar

TABLE 2
Predictive Performance of Each Comparing Approach (mean±std. deviation) in terms of *Average precision*, *Macro-averaging AUC* and *Adjusted hamming loss*. ↑ (↓) indicates the larger (smaller) the value, the better the performance. Best results are highlighted in **boldface**

| Data Sets | *Average precision* ↑ | | | | | | |
|---|---|---|---|---|---|---|---|
| | LIFT | LLSF | JFSC | TIFS | C2AE | MPVAE | CLIF |
| CAL500 | 0.500±0.015 | 0.511±0.016 | 0.502±0.012 | 0.471±0.019 | 0.478±0.014 | 0.480±0.013 | **0.513±0.016** |
| Image | 0.824±0.019 | 0.754±0.023 | 0.763±0.020 | 0.768±0.025 | 0.782±0.019 | 0.817±0.022 | **0.836±0.021** |
| scene | 0.886±0.016 | 0.847±0.016 | 0.853±0.021 | 0.828±0.011 | 0.851±0.019 | 0.879±0.022 | **0.888±0.016** |
| yeast | 0.770±0.017 | 0.763±0.014 | 0.759±0.013 | 0.751±0.023 | 0.733±0.019 | 0.764±0.013 | **0.773±0.018** |
| corel5k | 0.288±0.011 | 0.301±0.012 | 0.301±0.012 | 0.236±0.016 | 0.272±0.012 | 0.311±0.012 | **0.336±0.013** |
| rcv1-s1 | 0.596±0.010 | 0.620±0.010 | 0.620±0.012 | 0.488±0.021 | 0.621±0.015 | 0.639±0.012 | **0.646±0.013** |
| Corel16k-s1 | 0.320±0.005 | 0.346±0.007 | 0.345±0.007 | 0.245±0.004 | 0.333±0.007 | 0.355±0.008 | **0.369±0.008** |
| delicious | 0.378±0.005 | 0.362±0.005 | 0.381±0.005 | 0.257±0.008 | 0.357±0.004 | 0.400±0.005 | **0.403±0.005** |
| iaprtc12 | 0.346±0.005 | 0.368±0.005 | 0.373±0.005 | 0.291±0.007 | 0.372±0.006 | 0.411±0.005 | **0.420±0.005** |
| espgame | 0.284±0.005 | 0.277±0.004 | 0.279±0.005 | 0.210±0.004 | 0.276±0.004 | **0.311±0.005** | 0.308±0.004 |
| mirflickr | 0.635±0.003 | 0.651±0.006 | 0.651±0.006 | 0.542±0.007 | 0.655±0.005 | 0.661±0.005 | **0.671±0.006** |
| tmc2007 | 0.815±0.003 | 0.815±0.003 | 0.809±0.003 | 0.750±0.006 | 0.791±0.005 | **0.836±0.004** | 0.833±0.005 |
| mediamill | 0.730±0.003 | 0.728±0.003 | 0.712±0.004 | 0.634±0.007 | 0.721±0.003 | 0.747±0.003 | **0.752±0.004** |
| bookmarks | 0.492±0.004 | 0.501±0.002 | 0.499±0.002 | 0.307±0.003 | 0.489±0.003 | **0.520±0.002** | 0.508±0.003 |

| Data Sets | *Macro-averaging AUC* ↑ | | | | | | |
|---|---|---|---|---|---|---|---|
| | LIFT | LLSF | JFSC | TIFS | C2AE | MPVAE | CLIF |
| CAL500 | 0.518±0.011 | **0.579±0.016** | 0.552±0.012 | 0.542±0.020 | 0.487±0.025 | 0.496±0.013 | 0.567±0.016 |
| Image | 0.858±0.015 | 0.793±0.021 | 0.818±0.021 | 0.818±0.020 | 0.824±0.024 | 0.851±0.021 | **0.870±0.021** |
| scene | 0.948±0.009 | 0.921±0.011 | 0.924±0.012 | 0.915±0.007 | 0.919±0.017 | 0.946±0.011 | **0.950±0.010** |
| yeast | 0.675±0.019 | 0.694±0.016 | 0.678±0.016 | 0.674±0.023 | 0.625±0.023 | 0.705±0.012 | **0.715±0.016** |
| corel5k | 0.717±0.013 | 0.662±0.017 | 0.671±0.013 | 0.655±0.016 | 0.677±0.009 | 0.688±0.020 | **0.760±0.010** |
| rcv1-s1 | 0.926±0.007 | 0.912±0.009 | 0.907±0.011 | 0.849±0.016 | 0.918±0.006 | 0.937±0.005 | **0.946±0.004** |
| Corel16k-s1 | 0.688±0.008 | 0.710±0.006 | 0.708±0.009 | 0.652±0.015 | 0.732±0.009 | 0.738±0.014 | **0.787±0.005** |
| delicious | 0.782±0.004 | 0.766±0.005 | 0.772±0.006 | 0.641±0.010 | 0.790±0.004 | 0.813±0.004 | **0.827±0.003** |
| iaprtc12 | 0.798±0.005 | 0.816±0.005 | 0.808±0.006 | 0.786±0.006 | 0.842±0.003 | 0.857±0.003 | **0.863±0.004** |
| espgame | 0.761±0.007 | 0.738±0.006 | 0.728±0.007 | 0.656±0.008 | 0.761±0.007 | 0.779±0.005 | **0.781±0.004** |
| mirflickr | 0.797±0.005 | 0.821±0.005 | 0.810±0.005 | 0.709±0.007 | 0.817±0.006 | 0.823±0.007 | **0.833±0.004** |
| tmc2007 | 0.923±0.003 | 0.923±0.004 | 0.920±0.004 | 0.875±0.002 | 0.896±0.006 | 0.933±0.003 | **0.934±0.003** |
| mediamill | 0.774±0.011 | 0.778±0.004 | 0.834±0.009 | 0.707±0.007 | 0.800±0.011 | **0.861±0.009** | 0.859±0.009 |
| bookmarks | 0.894±0.002 | 0.882±0.003 | 0.873±0.003 | 0.679±0.011 | 0.859±0.004 | **0.912±0.003** | 0.906±0.003 |

| Data Sets | *Adjusted hamming loss* ↓ | | | | | | |
|---|---|---|---|---|---|---|---|
| | LIFT | LLSF | JFSC | TIFS | C2AE | MPVAE | CLIF |
| CAL500 | 0.678±0.009 | 0.660±0.009 | 0.687±0.007 | 0.622±0.016 | **0.553±0.008** | 0.668±0.007 | 0.575±0.013 |
| Image | 0.417±0.035 | 0.501±0.023 | 0.623±0.025 | 0.390±0.033 | 0.409±0.027 | 0.350±0.026 | **0.336±0.022** |
| scene | 0.291±0.027 | 0.525±0.037 | 0.449±0.032 | 0.303±0.025 | 0.317±0.029 | 0.237±0.029 | **0.235±0.023** |
| yeast | 0.376±0.018 | 0.387±0.017 | 0.377±0.017 | 0.355±0.015 | 0.395±0.021 | 0.352±0.014 | **0.346±0.017** |
| corel5k | 0.948±0.007 | 0.947±0.007 | 0.917±0.008 | 0.832±0.017 | 0.798±0.007 | 0.829±0.013 | **0.787±0.013** |
| rcv1-s1 | 0.726±0.014 | 0.723±0.016 | 0.672±0.016 | 0.623±0.017 | 0.498±0.011 | 0.460±0.011 | **0.459±0.007** |
| Corel16k-s1 | 0.969±0.005 | 0.956±0.004 | 0.937±0.007 | 0.838±0.005 | 0.770±0.006 | 0.784±0.008 | **0.729±0.008** |
| delicious | 0.780±0.006 | 0.812±0.004 | 0.798±0.004 | 0.761±0.005 | 0.747±0.002 | 0.729±0.005 | **0.698±0.007** |
| iaprtc12 | 0.882±0.003 | 0.894±0.004 | 0.904±0.004 | 0.843±0.004 | 0.775±0.005 | 0.645±0.004 | **0.639±0.007** |
| espgame | 0.939±0.005 | 0.958±0.004 | 0.960±0.004 | 0.918±0.003 | 0.819±0.004 | 0.825±0.007 | **0.763±0.007** |
| mirflickr | 0.622±0.006 | 0.606±0.006 | 0.631±0.006 | 0.594±0.008 | 0.499±0.005 | 0.495±0.006 | **0.488±0.007** |
| tmc2007 | 0.373±0.007 | 0.388±0.005 | 0.387±0.005 | 0.389±0.006 | 0.347±0.009 | **0.294±0.006** | 0.299±0.006 |
| mediamill | 0.452±0.003 | 0.463±0.003 | 0.490±0.004 | 0.504±0.006 | 0.445±0.003 | 0.418±0.004 | **0.406±0.005** |
| bookmarks | 0.742±0.002 | 0.805±0.002 | 0.801±0.003 | 0.816±0.003 | 0.646±0.004 | **0.617±0.001** | 0.637±0.004 |

- TIFS [11]: TIFS performs label-specific feature selection in a latent topic space which captures the input-output correlation. [parameter configuration: grid search for $\tau, \delta \in \{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$]
- C2AE [59]: C2AE is a deep neural network based label embedding approach for multi-label classification, which jointly embeds features and labels via integrating deep canonical correlation analysis and autoencoder. [parameter configuration: search for $\alpha \in \{0.1, 1, 2, 5, 10\}$]
- MPVAE [60]: MPVAE employs a variational autoencoder to align features and labels in a probabilistic latent space and explicitly learns a shared covariance matrix to model the label correlation. [parameter configuration: $\lambda_1 = \lambda_2 = 0.5$, $\lambda_3 = 10$, $\beta = 1.1$]

For CLIF, trade-off parameter $\lambda$ is searched in $\{10^{-5}, 10^{-4}, \ldots, 1, 2, 5, 10\}$ and the output dimensionality of GIN, i.e. $d_e$ is searched in $\{64, 128, 256\}$. For fair comparison, all neural network-based approaches share the same neural network structure. Ten-fold cross validation is employed to evaluate above approaches on the 14 benchmark data sets.

Table 2 and Table 3 report detailed experimental results of comparing algorithms in terms of each evaluation metric. For each evaluation metric, "↓" indicates "the smaller the better" while "↑" indicates "the larger the better". Furthermore, the best performance among comparing algorithms is shown in boldface.

To analyze whether there are statistical performance gaps among comparing algorithms, *Friedman test* [72], which

TABLE 3
Predictive Performance of Each Comparing Approach (mean±std. deviation) in terms of *One-error*, *Coverage* and *Ranking loss*. ↑ (↓) indicates the larger (smaller) the value, the better the performance. Best results are highlighted in **boldface**

| Data Sets | *One-error* ↓ | | | | | | |
|---|---|---|---|---|---|---|---|
| | LIFT | LLSF | JFSC | TIFS | C2AE | MPVAE | CLIF |
| CAL500 | 0.124±0.033 | 0.118±0.043 | **0.116±0.028** | 0.179±0.061 | **0.116±0.026** | 0.117±0.035 | 0.119±0.046 |
| Image | 0.269±0.033 | 0.380±0.039 | 0.369±0.035 | 0.363±0.041 | 0.332±0.029 | 0.279±0.038 | **0.252±0.032** |
| scene | 0.196±0.029 | 0.255±0.025 | 0.243±0.033 | 0.292±0.021 | 0.240±0.030 | 0.207±0.039 | **0.189±0.030** |
| yeast | **0.218±0.034** | **0.218±0.025** | 0.230±0.023 | 0.249±0.038 | 0.256±0.033 | 0.235±0.019 | 0.221±0.033 |
| corel5k | 0.682±0.012 | 0.645±0.018 | 0.638±0.018 | 0.747±0.031 | 0.660±0.017 | 0.632±0.024 | **0.609±0.018** |
| rcv1-s1 | 0.408±0.015 | 0.412±0.016 | 0.411±0.018 | 0.561±0.033 | 0.434±0.017 | 0.403±0.017 | **0.402±0.017** |
| Corel16k-s1 | 0.676±0.013 | 0.636±0.018 | 0.636±0.018 | 0.779±0.011 | 0.652±0.017 | 0.630±0.017 | **0.621±0.018** |
| delicious | 0.325±0.012 | 0.354±0.011 | 0.322±0.009 | 0.561±0.027 | 0.358±0.009 | 0.310±0.011 | **0.304±0.010** |
| iaprtc12 | 0.501±0.007 | 0.478±0.009 | 0.473±0.009 | 0.627±0.012 | 0.498±0.010 | 0.436±0.014 | **0.430±0.008** |
| espgame | 0.631±0.009 | 0.640±0.013 | 0.638±0.011 | 0.722±0.005 | 0.654±0.010 | 0.604±0.013 | **0.603±0.007** |
| mirflickr | 0.317±0.008 | 0.300±0.011 | 0.306±0.011 | 0.434±0.007 | 0.292±0.011 | 0.294±0.008 | **0.281±0.010** |
| tmc2007 | 0.217±0.004 | 0.223±0.008 | 0.230±0.007 | 0.286±0.011 | 0.235±0.012 | **0.196±0.006** | 0.198±0.006 |
| mediamill | 0.158±0.006 | 0.159±0.006 | 0.172±0.007 | 0.220±0.011 | 0.159±0.006 | 0.149±0.007 | **0.145±0.006** |
| bookmarks | 0.533±0.005 | 0.523±0.003 | 0.524±0.003 | 0.727±0.003 | 0.524±0.005 | **0.506±0.002** | 0.520±0.005 |

| Data Sets | *Coverage* ↓ | | | | | | |
|---|---|---|---|---|---|---|---|
| | LIFT | LLSF | JFSC | TIFS | C2AE | MPVAE | CLIF |
| CAL500 | 0.753±0.024 | 0.754±0.016 | **0.743±0.013** | 0.788±0.023 | 0.798±0.016 | 0.779±0.014 | 0.744±0.016 |
| Image | 0.169±0.013 | 0.222±0.019 | 0.209±0.016 | 0.207±0.018 | 0.204±0.018 | 0.174±0.017 | **0.161±0.018** |
| scene | 0.066±0.008 | 0.089±0.009 | 0.088±0.011 | 0.096±0.006 | 0.094±0.011 | 0.069±0.009 | **0.065±0.006** |
| yeast | 0.452±0.015 | 0.454±0.017 | 0.465±0.016 | 0.456±0.013 | 0.468±0.018 | 0.457±0.017 | **0.445±0.014** |
| corel5k | 0.291±0.012 | 0.436±0.014 | 0.444±0.017 | 0.284±0.016 | 0.331±0.012 | 0.247±0.015 | **0.219±0.013** |
| rcv1-s1 | 0.121±0.007 | 0.118±0.010 | 0.137±0.011 | 0.181±0.014 | 0.109±0.004 | 0.089±0.006 | **0.082±0.006** |
| Corel16k-s1 | 0.324±0.007 | 0.324±0.008 | 0.342±0.008 | 0.321±0.005 | 0.284±0.008 | 0.250±0.007 | **0.233±0.005** |
| delicious | 0.481±0.007 | 0.618±0.009 | 0.602±0.010 | 0.580±0.009 | 0.429±0.006 | 0.410±0.004 | **0.405±0.006** |
| iaprtc12 | 0.320±0.005 | 0.377±0.008 | 0.376±0.008 | 0.372±0.008 | 0.276±0.005 | 0.265±0.004 | **0.248±0.004** |
| espgame | 0.351±0.009 | 0.454±0.008 | 0.463±0.009 | 0.411±0.009 | 0.353±0.007 | 0.320±0.008 | **0.314±0.006** |
| mirflickr | 0.317±0.003 | 0.319±0.004 | 0.327±0.005 | 0.375±0.005 | 0.309±0.004 | 0.303±0.005 | **0.286±0.004** |
| tmc2007 | 0.121±0.004 | 0.127±0.004 | 0.131±0.004 | 0.159±0.003 | 0.148±0.005 | **0.112±0.003** | 0.113±0.003 |
| mediamill | 0.156±0.003 | 0.174±0.004 | 0.170±0.004 | 0.215±0.004 | 0.155±0.003 | 0.132±0.002 | **0.131±0.003** |
| bookmarks | 0.131±0.002 | 0.157±0.004 | 0.165±0.004 | 0.246±0.006 | 0.173±0.004 | **0.112±0.002** | 0.116±0.002 |

| Data Sets | *Ranking loss* ↓ | | | | | | |
|---|---|---|---|---|---|---|---|
| | LIFT | LLSF | JFSC | TIFS | C2AE | MPVAE | CLIF |
| CAL500 | 0.181±0.006 | 0.184±0.007 | **0.179±0.005** | 0.206±0.008 | 0.196±0.005 | 0.195±0.006 | **0.179±0.005** |
| Image | 0.143±0.014 | 0.212±0.023 | 0.193±0.019 | 0.193±0.021 | 0.189±0.024 | 0.148±0.020 | **0.134±0.022** |
| scene | **0.062±0.010** | 0.089±0.011 | 0.088±0.015 | 0.098±0.008 | 0.094±0.014 | 0.066±0.012 | **0.062±0.008** |
| yeast | 0.164±0.010 | 0.168±0.010 | 0.177±0.009 | 0.178±0.012 | 0.187±0.012 | 0.172±0.012 | **0.162±0.010** |
| corel5k | 0.122±0.005 | 0.191±0.008 | 0.196±0.008 | 0.131±0.006 | 0.161±0.006 | 0.105±0.005 | **0.098±0.005** |
| rcv1-s1 | 0.048±0.003 | 0.046±0.004 | 0.054±0.005 | 0.082±0.006 | 0.046±0.002 | 0.036±0.003 | **0.033±0.003** |
| Corel16k-s1 | 0.163±0.003 | 0.162±0.004 | 0.173±0.004 | 0.178±0.003 | 0.153±0.004 | 0.128±0.002 | **0.122±0.002** |
| delicious | 0.100±0.002 | 0.143±0.003 | 0.121±0.003 | 0.137±0.002 | 0.103±0.002 | 0.089±0.001 | **0.086±0.002** |
| iaprtc12 | 0.111±0.002 | 0.123±0.003 | 0.122±0.003 | 0.138±0.004 | 0.095±0.002 | 0.090±0.001 | **0.083±0.002** |
| espgame | 0.143±0.003 | 0.182±0.004 | 0.186±0.004 | 0.183±0.003 | 0.148±0.003 | 0.129±0.003 | **0.128±0.002** |
| mirflickr | 0.120±0.002 | 0.119±0.003 | 0.122±0.004 | 0.163±0.004 | 0.115±0.004 | 0.111±0.004 | **0.103±0.003** |
| tmc2007 | 0.047±0.002 | 0.049±0.002 | 0.051±0.002 | 0.075±0.002 | 0.062±0.003 | **0.040±0.001** | **0.040±0.002** |
| mediamill | 0.045±0.001 | 0.052±0.002 | 0.052±0.002 | 0.067±0.002 | 0.046±0.001 | **0.037±0.001** | **0.037±0.001** |
| bookmarks | 0.083±0.001 | 0.098±0.003 | 0.102±0.003 | 0.179±0.005 | 0.113±0.003 | **0.071±0.001** | 0.073±0.002 |

TABLE 4
Summary of the Friedman Statistics $F_F$ in terms of Each Evaluation Metric and the Critical Value at 0.05 Significance Level (# comparing algorithms $K = 7$, # data sets $N = 14$)

| Evaluation metric | $F_F$ | Critical value ($\alpha = 0.05$) |
|---|---|---|
| *Average precision* | 33.493 | 2.217 |
| *Macro-averaging AUC* | 24.708 | |
| *Adjusted hamming loss* | 33.028 | |
| *One-error* | 20.230 | |
| *Coverage* | 20.949 | |
| *Ranking loss* | 31.176 | |

is a widely-accepted statistical test for comparisons of multiple algorithms over a number of data sets, is employed. For each evaluation metric, the average rank of the $j$-th algorithm is firstly computed as $R_j = \frac{1}{N}\sum_{i=1}^{N} r_i^j$, where $r_i^j$ denotes the rank of the $j$-th algorithm on the $i$-th data set. Then, the Friedman statistics $F_F$, which is distributed according to the $F$-distribution with $(K-1)$ numerator degrees of freedom and $(K-1)(N-1)$ denominator degrees of freedom, is computed as:

$$F_F = \frac{(N-1)\mathcal{X}_F^2}{N(K-1)-\mathcal{X}_F^2}, \quad where$$

$$\mathcal{X}_F^2 = \frac{12N}{K(K+1)}[\sum_{j=1}^{K} R_j^2 - \frac{K(K+1)^2}{4}]$$

Table 4 summarizes the Friedman statistics $F_F$ on each evaluation metric and the corresponding critical value at significance level $\alpha = 0.05$. As shown in Table 4, the

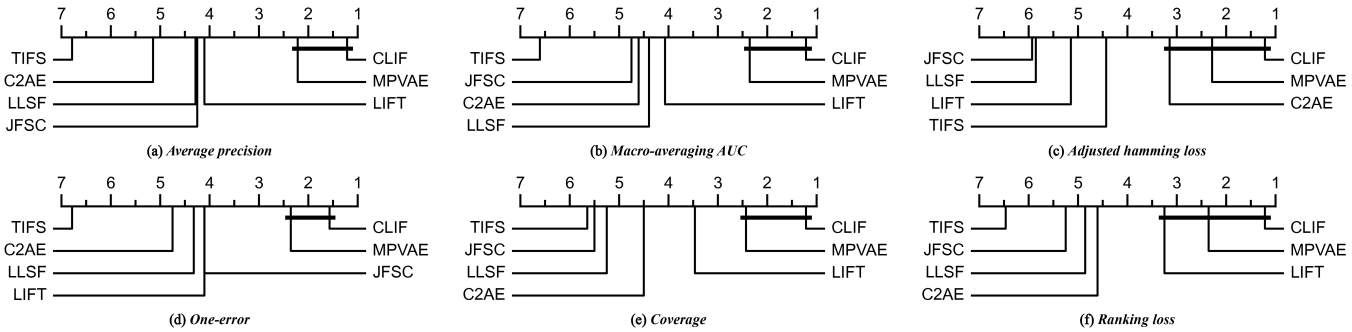


Fig. 2. Comparison of CLIF (control algorithm) against other comparing algorithms with the *Bonferroni-Dunn test*. Algorithms not connected with CLIF in the CD diagram are considered to have significantly different performance from the control algorithm (significance level $\alpha = 0.05$).

null hypothesis, i.e. all comparing algorithms possess equal performance, is clearly rejected in terms of each evaluation metric.

Consequently, the *Bonferroni-Dunn test* [73] is further employed as the *post-hoc test* [72] to analyze the relative performance among the comparing approaches. Here, CLIF is treated as the control algorithm and the difference between the average ranks of CLIF and one comparing algorithm is compared with the *critical difference* (CD). If their difference is larger than one CD (CD=2.154 with $K = 7$ and $N = 14$ at significance level of $\alpha = 0.05$), the performance of CLIF is deemed to be significantly different from that of the comparing algorithm.

Fig. 2 presents the CD diagrams [72] on each evaluation metric. In each subfigure, the average rank of each comparing algorithm is marked along the axis with lower ranks to the right and a thick line connects CLIF and any comparing algorithm if the difference between their average ranks is less than one CD. Based on the above results, observations can be made as follows:

- CLIF achieves lowest average ranks against other comparing approaches in terms of each evaluation metric. Furthermore, across all evaluation metrics, CLIF ranks 1st in 79.8% cases over all the 14 data sets.
- As shown in Fig. 2, CLIF performs better than other approaches based on label-specific features. Concretely, CLIF significantly outperforms LLSF, JFSC and TIFS in terms of all evaluation metrics and achieves statistically superior or at least comparable performance against LIFT. These impressive results validate the superiority of our collaborative learning strategy against existing approaches based on label-specific features.
- Furthermore, Fig. 2 shows that CLIF significantly outperforms C2AE in terms of all evaluation metrics except for Adjusted hamming loss, while CLIF is not significantly different from MPVAE. This is due to the factor that MPVAE beats other comparing approaches and the Bonferroni-Dunn test fails to detect that CLIF achieves a consistently better average ranks than MPVAE on all evaluation metrics. The superior performance of CLIF against C2AE and MPVAE indicates that it is a promising direction to explore the interactions between the feature space and the label space via extracting label-specific features under the guidance of collaboratively learned label semantics.

To summarize, CLIF achieves highly competitive performance against other well-established multi-label classification algorithms, which validates the effectiveness of our proposed label-specific feature learning approach to facilitate multi-label classification.

### 4.3 Further Analyses

#### 4.3.1 Ablation Studies

In this section, ablation studies are conducted on all the 14 multi-label benchmark data sets with 10-fold cross validation. The training and test settings of CLIF's variant models are exactly the same as those of CLIF unless otherwise stated. We conduct the *Wilcoxon signed-ranks test* [74] at significance level $\alpha = 0.05$ to analyze whether CLIF performs statistically better than these variant models. Table 5 summarizes the $p$-value statistics on each evaluation metric and Table 6 shows the detailed experimental results in terms of Average precision.

*Effectiveness of the collaborative learning strategy.* We implement a variant model named CLIF-ts, which learns label semantics and label-specific features in a two-stage training procedure. Concretely, CLIF-ts firstly learns to encode semantic relations among labels into semantic label embeddings with the label-embedding loss $\mathcal{L}_{le}$ elaborated in Eq. (2). Then, CLIF-ts learns to extract label-specific features and perform classification with the cross-entropy loss $\mathcal{L}_{ce}$, freezing the learned semantic label embeddings. Results reported in Table 5 validate the effectiveness of our collaborative learning strategy.

*Effectiveness of the semantic-guided label-specific features.* We implement a plain version of CLIF named CLIF-id, where identical features are employed in the discrimination processes of all class labels. As shown in Table 5, CLIF achieves comparable performance against the CLIF-id variation in terms of Adjusted hamming loss and significantly outperforms it on all the other evaluation metrics.

*Effectiveness of the label semantic encoding module.* We implement two variant models named CLIF-re and CLIF-oe. CLIF-re is implemented by replacing the label semantic encoding module with label embedding matrix generated by Gaussian function with zero mean and standard deviation of 1, while CLIF-oe is implemented similarly but with a one-hot label embedding matrix. As shown in Table 5, the

TABLE 5
Summary of the Wilcoxon Signed-Ranks Test for CLIF Against Its Variants in terms of Each Evaluation Metric at 0.05 Significance Level. $p$-values are Shown in the Brackets

| Comparing approaches | *Average precision* | *Macro-averaging AUC* | *Adjusted hamming loss* | *One-error* | *Coverage* | *Ranking loss* |
|---|---|---|---|---|---|---|
| CLIF against CLIF-ts | **win** [0.0004] | **win** [0.0205] | **win** [0.0026] | **win** [0.0002] | **win** [0.0039] | **win** [0.0020] |
| CLIF against CLIF-id | **win** [0.0020] | **win** [0.0352] | **tie** [0.0762] | **win** [0.0291] | **win** [0.0024] | **win** [0.0391] |
| CLIF against CLIF-re | **win** [0.0001] | **win** [0.0001] | **win** [0.0006] | **win** [0.0200] | **win** [0.0002] | **win** [0.0059] |
| CLIF against CLIF-oe | **win** [0.0010] | **tie** [0.0715] | **win** [0.0002] | **win** [0.0001] | **win** [0.0103] | **win** [0.0093] |

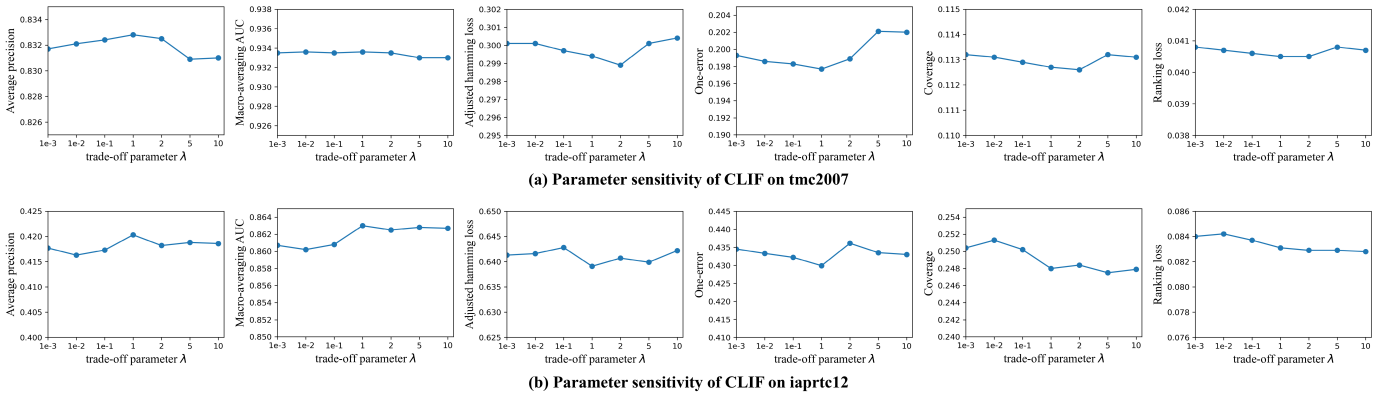


Fig. 3. Performance of CLIF changes as the trade-off parameter $\lambda$ varies. The first and the second rows show results on the *tmc2007* and *iaprtc12* data sets respectively.

TABLE 6
Predictive Performance of CLIF and its variant models (mean±std. deviation) in terms of *Average precision*

| Data Sets | *Average precision* ↑ | | | | |
|---|---|---|---|---|---|
| | CLIF | CLIF-ts | CLIF-id | CLIF-re | CLIF-oe |
| CAL500 | **0.513±0.016** | 0.511±0.017 | 0.509±0.015 | 0.510±0.015 | 0.510±0.015 |
| Image | **0.836±0.021** | 0.828±0.022 | 0.830±0.020 | 0.829±0.022 | 0.828±0.019 |
| scene | **0.888±0.016** | 0.884±0.019 | **0.888±0.019** | 0.883±0.022 | 0.884±0.022 |
| yeast | **0.773±0.018** | 0.771±0.017 | 0.772±0.017 | 0.771±0.017 | 0.770±0.015 |
| corel5k | **0.336±0.013** | 0.330±0.012 | 0.335±0.014 | 0.333±0.012 | 0.332±0.013 |
| rcv1-s1 | **0.646±0.013** | 0.636±0.012 | 0.633±0.014 | 0.645±0.011 | 0.630±0.012 |
| Corel16k-s1 | 0.369±0.008 | 0.368±0.006 | 0.368±0.007 | 0.367±0.007 | **0.370±0.007** |
| delicious | **0.403±0.005** | 0.393±0.004 | **0.403±0.005** | 0.395±0.005 | 0.391±0.005 |
| iaprtc12 | 0.420±0.005 | **0.421±0.007** | 0.411±0.005 | 0.419±0.006 | **0.421±0.006** |
| espgame | **0.308±0.004** | 0.305±0.005 | **0.308±0.004** | 0.303±0.005 | **0.308±0.006** |
| mirflickr | **0.671±0.006** | 0.664±0.007 | 0.659±0.012 | 0.666±0.007 | 0.667±0.008 |
| tmc2007 | **0.833±0.005** | 0.829±0.004 | 0.831±0.005 | 0.829±0.005 | 0.829±0.005 |
| mediamill | **0.752±0.004** | 0.745±0.003 | 0.751±0.004 | 0.750±0.004 | 0.744±0.004 |
| bookmarks | **0.508±0.003** | 0.499±0.003 | **0.508±0.002** | 0.507±0.003 | 0.500±0.003 |

effectiveness of the label semantic encoding module is statistically significant. In CLIF, label embeddings generated by the label semantic encoding module capture semantics from label space. Instead, label embedding matrix generated by Gaussian function or one-hot encoder lacks such semantics.

### 4.3.2 Parameter Sensitivity

Fig. 3 gives an illustrative example on how the performance of CLIF changes in terms of each evaluation metric when the value of the trade-off parameter $\lambda$ in the overall objective function changes. As shown in Fig. 3, the trade-off parameter $\lambda$ which controls the strength of the preservation of label space topological structure does affect the performance of CLIF. However, the performance is still relatively stable as the parameter value changes within a reasonable range,

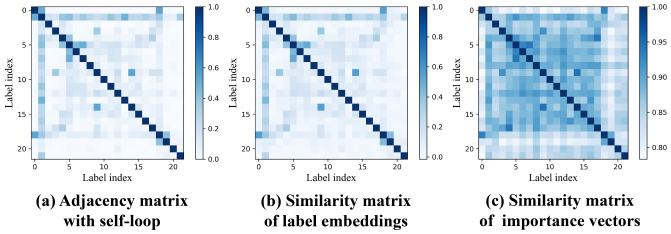


Fig. 4. Visualization of CLIF on tmc2007. The left subfigure shows the constructed adjacency matrix $\mathbf{A}$ with self-loop, i.e. filling the diagonal elements with 1. The middle and the right subfigures show the cosine similarity matrices of label embeddings and importance vectors respectively.

which serves as a desirable property in using the proposed approach. Similar results can be observed on other data sets.

### 4.3.3 Visualization

To provide a deeper insight into CLIF, we visualize the constructed adjacency matrix $\mathbf{A}$, learned semantic label embeddings and generated label-specific importance vectors $\alpha_k$ on the tmc2007 data set. It is obvious from Fig. 4 that: (a) the topological structure of the label space is well preserved in learned semantic label embeddings; (b) strongly correlated labels share more pertinent and discriminative features than weakly correlated labels. These results verify that CLIF can take full advantage of the semantics from label space to guide the learning process of label-specific features.

### 4.3.4 Complexity Analyses

Let $b$ be the batch size and $\hat{d}$ denote a proxy of the hidden dimensionalities of the network, the time complexity of CLIF corresponds to $\mathcal{O}(q^2\hat{d} + bq\hat{d}^2)$, where the quadratic term

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2021.3136592, IEEE Transactions on Pattern Analysis and Machine Intelligence



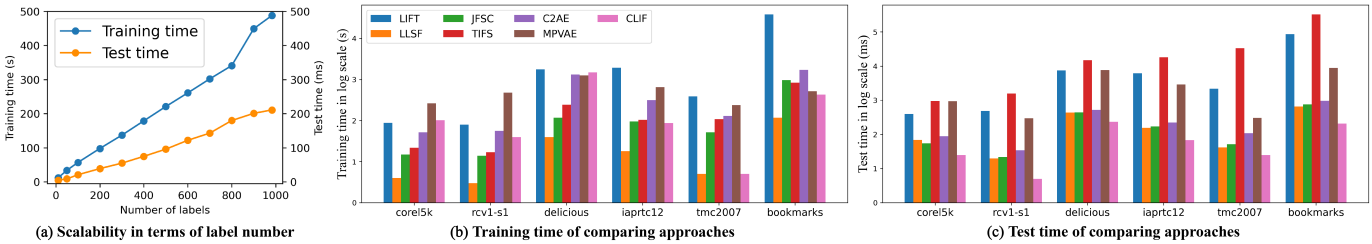


Fig. 5. Complexity analyses of CLIF. (a) Empirical scalability of CLIF on the *delicious* data set. (b)(c) Running time (training/test) of each comparing approach on six benchmark data sets. For histogram illustration, the $y$-axis corresponds to the logarithm of running time.

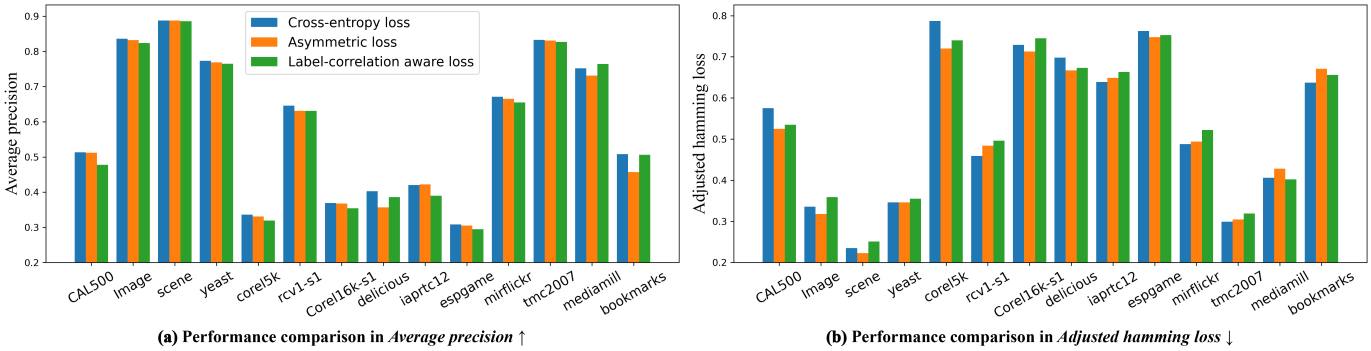


Fig. 6. Further analyses of CLIF with varying loss functions. (a) Performance comparision in terms of a ranking-based metric (*Average precision*). (b) Performance comparision in terms of a classification-based metric (*Adjusted hamming loss*).

of the label number $q$ is derived from the neighborhood aggregation process in Eq. (1). During test, this quadractic term can be eliminated by saving the final semantic label embeddings $\mathbf{E}$. As shown in Fig. 5(a), the actual time overhead of CLIF scales linearly in terms of $q$, when $q \ll b\hat{d}$. If $q$ becomes extremely large, e.g. millions of labels, this quadratic term can be optimized with approximate neighborhood aggregation [75] or hierarchical classification mechanism [48], [63], which will be left for further work.

Furthermore, Fig. 5(b)(c) illustrate the training and test time of each comparing approach, which show that CLIF is comparable to existing approaches in time overhead. On the same computing device (a V100 GPU), CLIF takes significantly less time to train and test than its strong competitor MPVAE.

#### 4.3.5 Alternative Implementations

As shown in Eq. (3), the default loss function for classification model induction in CLIF is the commonly-used cross-entropy loss. Here, we further analyze how the performance of CLIF changes when the loss funtion changes. Except for the cross-entropy loss, two more loss functions are investigated. The label-correlation aware loss [56] penalizes pairwise reversed ranking between each relevant-irrelevant label pair. While the asymmetric loss [76] concentrates on individual label discrimination as the cross-entropy loss does, but focuses more on hard relevant labels. As shown in Fig. 6, it is not surprising that the asymmetric loss tends to possess better performance in terms of *Adjusted hamming loss*, which is a classification-based metric. And the cross-entropy loss achieves better ranking performance, though the label-correlation aware loss is intuitively friendly to ranking-based metrics.

## 5 Conclusion

In this paper, we propose to construct label-specific features for multi-label classification with a novel collaborative learning strategy where the learning of label semantics and label-specific features interact and facilitate with each other. Following this strategy, we present a DNN-based approach CLIF which exploits learned label semantics to guide extracting the most pertinent features for each class label, while the discrimination process based on these label-specific features propagates label-specific discriminative properties to the learning process of the label semantics. Comprehensive experiments show that our approach achieves highly competitive performance against other well-established multi-label classification algorithms. In the future, it is interesting to design other interaction mechanisms between label semantics and label-specific features as there may be sophisticated relationships between a label and its own discriminative features.

## Acknowledgments

The authors wish to thank the associate editor and anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Science Foundation of China (62176055). We thank the Big Data Center of Southeast University for providing the facility support on the numerical calculations in this paper.

## References

[1] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1819–1837, 2014.

[2] A. Vallet and H. Sakamoto, "A multi-label convolutional neural network for automatic image annotation," *Journal of Information Processing*, vol. 23, no. 6, pp. 767–775, 2015.

[3] R. Cabral, F. Torre, J. Costeira, and A. Bernardino, "Matrix completion for multi-label image classification," in *Advances in Neural Information Processing Systems 24*, Granada, Spain, 2011, pp. 190–198.

[4] T. Rubin, A. Chambers, P. Smyth, and M. Steyvers, "Statistical topic models for multi-label document classification," *Machine learning*, vol. 88, no. 1-2, pp. 157–208, 2012.

[5] A. McCallum, "Multi-label text classification with a mixture model trained by EM," in *AAAI'99 Workshop on Text Learning*, Orlando, FL, 1999.

[6] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, "Multi-label classification of music by emotion," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2011, no. 1, pp. 1–9, 2011.

[7] B. Wu, E.-H. Zhong, A. Horner, and Q. Yang, "Music emotion recognition by multi-label multi-layer multi-instance multi-view learning," in *Proceedings of the 6th ACM International Conference on Multimedia*, Orlando, FL, 2014, pp. 117–126.

[8] M.-L. Zhang and L. Wu, "LIFT: Multi-label learning with label-specific features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 1, pp. 107–120, 2015.

[9] J. Huang, G.-R. Li, Q.-M. Huang, and X.-D. Wu, "Learning label specific features for multi-label classification," in *Proceedings of the 15th IEEE International Conference on Data Mining*, Atlantic City, NJ, 2015, pp. 181–190.

[10] X.-Y. Jia, S.-S. Zhu, and W.-W. Li, "Joint label-specific features and correlation information for multi-label learning," *Journal of Computer Science and Technology*, vol. 35, no. 2, pp. 247–258, 2020.

[11] J.-H. Ma and T. Chow, "Topic-based instance and feature selection in multilabel classification," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–15, 2020.

[12] X.-Y. Che, D.-G. Chen, and J.-S. Mi, "A novel approach for learning label correlation with application to feature selection of multi-label data," *Information Sciences*, vol. 512, pp. 795–812, 2020.

[13] H.-R. Han, M.-X. Huang, Y. Zhang, X.-G. Yang, and W.-L. Feng, "Multi-label learning with label specific features using correlation information," *IEEE Access*, vol. 7, pp. 11 474–11 484, 2019.

[14] Z.-S. Chen and M.-L. Zhang, "Multi-label learning with regularization enriched label-specific features," in *Proceedings of the 11th Asian Conference on Machine Learning*, Nagoya, Japan, 2019, pp. 411–424.

[15] J. Huang, G.-R. Li, Q.-M. Huang, and X.-D. Wu, "Learning label-specific features and class-dependent labels for multi-label classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 12, pp. 3309–3323, 2016.

[16] ——, "Joint feature selection and classification for multilabel learning," *IEEE Transactions on Cybernetics*, vol. 48, no. 3, pp. 876–889, 2018.

[17] J. Xu, H. Tian, Z. Wang, Y. Wang, W. Kang, and F. Chen, "Joint input and output space learning for multi-label image classification," *IEEE Transactions on Multimedia*, vol. 23, pp. 1696–1707, 2021.

[18] T. Hartvigsen, C. Sen, X.-N. Kong, and E. Rundensteiner, "Recurrent halting chain for early multi-label classification," in *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Virtual Event, CA, 2020, pp. 1382–1392.

[19] J. Zhang and X.-D. Wu, "Multi-label inference for crowdsourcing," in *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, London, UK, 2018, pp. 2738–2747.

[20] G. Tsoumakas, I. Katakis, and I. Vlahavas, *Mining Multi-Label Data*. Berlin, Germany: Springer, 2009, pp. 667–685.

[21] N. Xu, Y.-P. Liu, and X. Geng, "Label enhancement for label distribution learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 4, pp. 1632–1643, 2021.

[22] M.-L. Zhang, Q.-W. Zhang, J.-P. Fang, Y.-K. Li, and X. Geng, "Leveraging implicit relative labeling-importance information for effective multi-label learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 5, pp. 2057–2070, 2021.

[23] M. Boutell, J.-B. Luo, X.-P. Shen, and C. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.

[24] M.-L. Zhang, Y.-K. Li, X.-Y. Liu, and X. Geng, "Binary relevance for multi-label learning: An overview," *Frontiers of Computer Science*, vol. 12, no. 2, pp. 191–202, 2018.

[25] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Advances in Neural Information Processing Systems 14*, Vancouver, British Columbia, Canada, 2001, pp. 681–687.

[26] E. L. Mencía and J. Fürnkranz, "Pairwise learning of multilabel classifications with perceptrons," in *Proceedings of the International Joint Conference on Neural Networks*, Hong Kong, China, 2008, pp. 2899–2906.

[27] C. Brinker, E. L. Mencía, and J. Fürnkranz, "Graded multilabel classification by pairwise comparisons," in *Proceedings of the 14th IEEE International Conference on Data Mining*, Shenzhen, China, 2014, pp. 731–736.

[28] Y.-H. Guo and S.-C. Gu, "Multi-label classification using conditional dependency networks," in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, Barcelona, Catalonia, Spain, 2011, pp. 1300–1305.

[29] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine Learning*, vol. 85, no. 3, pp. 333–359, 2011.

[30] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Random k-labelsets for multilabel classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 7, pp. 1079–1089, 2010.

[31] T.-Z. Yu and W.-S. Zhang, "Semisupervised multilabel learning with joint dimensionality reduction," *IEEE Signal Processing Letters*, vol. 23, no. 6, pp. 795–799, 2016.

[32] R. B. Pereira, A. Plastino, B. Zadrozny, and L. H. C. Merschmann, "Categorizing feature selection methods for multi-label classification," *Artificial Intelligence Review*, vol. 49, no. 1, pp. 57–78, 2018.

[33] K. Bhatia, H. Jain, P. Kar, M. Varma, and P. Jain, "Sparse local embeddings for extreme multi-label classification," in *Advances in Neural Information Processing Systems 28*, Montreal, Quebec, Canada, 2015, pp. 730–738.

[34] Y.-N. Chen and H.-T. Lin, "Feature-aware label space dimension reduction for multi-label classification," in *Advances in Neural Information Processing Systems 25*, Lake Tahoe, NV, 2012, pp. 1538–1546.

[35] Y.-P. Sun and M.-L. Zhang, "Compositional metric learning for multi-label classification," *Frontiers of Computer Science*, vol. 15, no. 5, p. Article 155320, 2021.

[36] X. Wu, Q.-G. Chen, Y. Hu, D. Wang, X. Chang, X. Wang, and M.-L. Zhang, "Multi-view multi-label learning with view-specific information extraction," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, Macao, China, 2019, pp. 3884–3890.

[37] C. Zhang, Z. Yu, Q. Hu, P. Zhu, X. Liu, and X. Wang, "Latent semantic aware multi-view multi-label classification," in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, New Orleans, LA, 2018, pp. 4414–4421.

[38] S. D. Canuto, M. A. Gonçalves, and F. Benevenuto, "Exploiting new sentiment-based meta-level features for effective sentiment analysis," in *Proceedings of the 9th ACM International Conference on Web Search and Data Mining*, San Francisco, CA, 2016, pp. 53–62.

[39] Y. Yang and S. Gopal, "Multilabel classification with meta-level features in a learning-to-rank framework," *Machine Learning*, vol. 88, no. 1-2, pp. 47–68, 2012.

[40] S.-P. Xu, X.-B. Yang, H.-L. Yu, D.-J. Yu, J.-Y. Yang, and E. Tsang, "Multi-label learning with label-specific feature reduction," *Knowledge-Based Systems*, vol. 104, pp. 52–61, 2016.

[41] C. Zhang and Z. Li, "Multi-label learning with label-specific features via weighting and label entropy guided clustering ensemble," *Neurocomputing*, vol. 419, pp. 59–69, 2021.

[42] W. Zhan and M.-L. Zhang, "Multi-label learning with label-specific features via clustering ensemble," in *Proceedings of the 4th IEEE International Conference on Data Science and Advanced Analytics*, Tokyo, Japan, 2017, pp. 129–136.

[43] W. Weng, Y. Lin, S. Wu, Y. Li, and Y. Kang, "Multi-label learning based on label-specific features and local pairwise label correlation," *Neurocomputing*, vol. 273, pp. 385–394, 2018.

[44] Y. Guo, F. Chung, G. Li, J. Wang, and J. C. Gee, "Leveraging label-specific discriminant mapping features for multi-label learning," *ACM Transactions on Knowledge Discovery from Data*, vol. 13, no. 2, pp. 24:1–24:23, 2019.

[45] Z.-B. Yu and M.-L. Zhang, "Multi-label classification with label-specific feature generation: A wrapped approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[46] L. Sun, M. Kudo, and K. Kimura, "Multi-label classification with meta-label-specific features," in *Proceedings of the 23rd International*

*Conference on Pattern Recognition*, Cancún, Mexico, 2016, pp. 1612–1617.

[47] W. Weng, Y.-N. Chen, C.-L. Chen, S. Wu, and J. Liu, "Non-sparse label specific features selection for multi-label classification," *Neurocomputing*, vol. 377, pp. 85–94, 2020.

[48] R. You, Z. Zhang, Z. Wang, S. Dai, H. Mamitsuka, and S. Zhu, "AttentionXML: Label tree-based attention-aware deep model for high-performance extreme multi-label text classification," in *Advances in Neural Information Processing Systems 32*, Vancouver, Canada, 2019, pp. 5812–5822.

[49] T. Wei, W. Tu, and Y. Li, "Learning for tail label data: A label-specific feature approach," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, Macao, China, 2019, pp. 3842–3848.

[50] K. Wang, M. Yang, W. Yang, and Y. Yin, "Deep correlation structure preserved label space embedding for multi-label classification," in *Proceedings of the 10th Asian Conference on Machine Learning*, Beijing, China, 2018, pp. 1–16.

[51] L. Yang, X.-Z. Wu, Y. Jiang, and Z.-H. Zhou, "Multi-label learning with deep forest," in *Proceedings of the 24th European Conference on Artificial Intelligence*, vol. 325, Santiago de Compostela, Spain, 2020, pp. 1634–1641.

[52] T. Chen, L. Lin, X. Hui, R. Chen, and H. Wu, "Knowledge-guided multi-label few-shot learning for general image recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[53] J. Li, C. Zhang, J. T. Zhou, H. Fu, S. Xia, and Q. Hu, "Deep-LIFT: Deep label-specific feature learning for image annotation," *IEEE Transactions on Cybernetics*, pp. 1–10, 2021.

[54] M. Xu and L.-Z. Guo, "Learning from group supervision: the impact of supervision deficiency on multi-label learning," *Science China Information Sciences*, vol. 64, no. 3, p. Article 130101, 2021.

[55] H.-M. Chu, C.-K. Yeh, and Y.-C. F. Wang, "Deep generative models for weakly-supervised multi-label classification," in *Proceedings of the 15th European Conference on Computer Vision*, Munich, Germany, 2018, pp. 409–425.

[56] M.-L. Zhang and Z.-H. Zhou, "Multi-label neural networks with applications to functional genomics and text categorization," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 10, pp. 1338–1351, 2006.

[57] C. Chen, H.-B. Wang, W.-W. Liu, X.-Y. Zhao, T.-L. Hu, and G. Chen, "Two-stage label embedding via neural factorization machine for multi-label classification," in *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, Honolulu, HI, 2019, pp. 3304–3311.

[58] X. Shen, W. Liu, Y. Luo, Y.-S. Ong, and I. W. Tsang, "Deep discrete prototype multilabel learning," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, Stockholm, Sweden, 2018, pp. 2675–2681.

[59] C. Yeh, W. Wu, W. Ko, and Y. Wang, "Learning deep latent spaces for multi-label classification," in *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, San Francisco, CA, 2017, pp. 2838–2844.

[60] J.-W. Bai, S.-F. Kong, and C. Gomes, "Disentangled variational autoencoder based multi-label classification with covariance-aware multivariate probit model," in *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, Yokohama, Japan, 2020, pp. 4313–4321.

[61] J. Lanchantin, A. Sekhon, and Y. Qi, "Neural message passing for multi-label classification," in *Machine Learning and Knowledge Discovery in Databases*, vol. 11907, Würzburg, Germany, 2019, pp. 138–163.

[62] D. Chen, Y. Xue, and C. P. Gomes, "End-to-end learning for the deep multivariate probit model," in *Proceedings of the 35th International Conference on Machine Learning*, Stockholm, Sweden, 2018, pp. 931–940.

[63] A. Mittal, N. Sachdeva, S. Agrawal, S. Agarwal, P. Kar, and M. Varma, "ECLARE: Extreme classification with label graph correlations," in *Proceedings of the Web Conference*, Ljubljana, Slovenia, 2021, pp. 3721–3732.

[64] Z.-M. Chen, X.-S. Wei, P. Wang, and Y.-W. Guo, "Multi-label image recognition with graph convolutional networks," in *Proceedings of the 32nd IEEE Conference on Computer Vision and Pattern Recognition*, Long Beach, CA, 2019, pp. 5177–5186.

[65] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu, "CNN-RNN: A unified framework for multi-label image classification," in *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, 2016, pp. 2285–2294.

[66] V. O. Yazici, A. Gonzalez-Garcia, A. Ramisa, B. Twardowski, and J. van de Weijer, "Orderless recurrent models for multi-label classification," in *Proceedings of the 33rd IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, 2020, pp. 13 437–13 446.

[67] K.-Y. Xu, W.-H. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *Proceedings of the 7th International Conference on Learning Representations*, New Orleans, LA, 2019.

[68] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, Lille, France, 2015, pp. 448–456.

[69] A. Maas, A. Hannun, and A. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proceedings of the 30th International Conference on Machine Learning*, Atlanta, GA, 2013, pp. 1–6.

[70] F. Charte, A. J. Rivera, M. J. del Jesus, and F. Herrera, "Mlenn: A first approach to heuristic multilabel undersampling," in *Proceedings of the 15th International Conference on Intelligent Data Engineering and Automated Learning*, vol. 8669, Salamanca, Spain, 2014, pp. 1–9.

[71] K.-M. He, X.-Y. Zhang, S.-Q. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of International Conference on Computer Vision*, Santiago, Chile, 2015, pp. 1026–1034.

[72] J. Demsar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[73] O. Dunn, "Multiple comparisons among means," *Journal of the American statistical association*, vol. 56, no. 293, pp. 52–64, 1961.

[74] F. Wilcoxon, *Individual Comparisons by Ranking Methods*. Berlin, Germany: Springer, 1992, pp. 196–202.

[75] W. L. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Advances in Neural Information Processing Systems 30*, Long Beach, CA, 2017, pp. 1024–1034.

[76] T. Ridnik, E. Ben-Baruch, N. Zamir, A. Noy, I. Friedman, M. Protter, and L. Zelnik-Manor, "Asymmetric loss for multi-label classification," in *Proceedings of the IEEE International Conference on Computer Vision*, Virtual, 2021, pp. 82–91.

**Jun-Yi Hang** received the BSc and MSc degrees from Beihang University, China, in 2017 and 2020 respectively. Currently, he is a PhD student at the School of Computer Science and Engineering, Southeast University, China. His main research interests include machine learning and data mining, especially in learning from multi-label data.

**Min-Ling Zhang** received the BSc, MSc, and PhD degrees in computer science from Nanjing University, China, in 2001, 2004 and 2007, respectively. Currently, he is a Professor at the School of Computer Science and Engineering, Southeast University, China. His main research interests include machine learning and data mining. In recent years, Dr. Zhang has served as the General Co-Chairs of ACML'18, Program Co-Chairs of CCDM'20, PAKDD'19, CCF-ICAI'19, ACML'17, CCFAI'17, PRICAI'16, Senior PC member or Area Chair of AAAI 2017-2022, IJCAI 2017-2022, KDD 2021, ICDM 2015-2021, etc. He is also on the editorial board of IEEE Transactions on Pattern Analysis and Machine Intelligence, ACM Transactions on Intelligent Systems and Technology, Neural Networks, Science China Information Sciences, Frontiers of Computer Science, etc. Dr. Zhang is the Steering Committee Member of ACML and PAKDD, Vice Chair of the CAAI Machine Learning Society, standing committee member of the CCF Artificial Intelligence & Pattern Recognition Society. He is a Distinguished Member of CCF, CAAI, and Senior Member of ACM, IEEE.