Revisiting Pseudo-Label for Single-Positive Multi-Label Learning

Biao Liu [1] **Ning Xu** [1] **Jiaqi Lv** [2] **Xin Geng** [1]

Abstract

To deal with the challenge of high cost of annotating all relevant labels for each example in multi-label learning, single-positive multi-label learning (SPMLL) has been studied in recent years, where each example is annotated with only one positive label. By adopting *pseudo-label generation*, i.e., assigning pseudo-label to each example by various strategies, existing methods have empirically validated that SPMLL would significantly reduce the amount of supervision with a tolerable damage in classification performance. However, there is no existing method that can provide a theoretical guarantee for learning from pseudo-label on SPMLL. In this paper, the conditions of the effectiveness of learning from pseudo-label for SPMLL are shown and the learnability of pseudo-label-based methods is proven. Furthermore, based on the theoretical guarantee of pseudo-label for SPMLL, we propose a novel SPMLL method named MIME, i.e., Mutual label enhancement for sIngle-positive Multi-label lEarning and prove that the generated pseudo-label by MIME approximately converges to the fully-supervised case. Experiments on four image datasets and five MLL datasets show the effectiveness of our methods over several existing SPMLL approaches.

1. Introduction

Multi-label learning (MLL) aims to train a model on the examples that are associated with multiple labels and obtain a predictive model that is able to predict the relevant labels for an unknown instance accurately (Zhang & Zhou, 2013; Liu et al., 2021). Multi-label learning has been successfully applied to a variety of real-world applications during the past decade, such as image annotation (Wang et al., 2009), text classification (Liu et al., 2017) and facial expression recognition (Chen et al., 2020).

Comparing with multi-class-single-label learning, where an example is associated with only one positive label, multi-label learning requires a complete positive label set for each example. However, it is extremely difficult to accurately annotate each label of an example when the number of examples or categories is large. To deal with the challenge of high annotation cost, single-positive multi-label learning (SPMLL) (Cole et al., 2021; Xu et al., 2022) is proposed, where each example is annotated with only one positive label. Additionally, as many examples contain multiple categories but the annotation is a single label in multi-class datasets such as ImageNet (Yun et al., 2021), SPMLL would obtain multi-label predictors on existing numerous multi-class datasets, which could enhance the application of MLL.

Comparing with the fully labeled case, SPMLL is a more challenging problem, where a model trained with the single positive labels would collapse to a trivial solution, i.e., the model tends to predict every label as a positive one. To alleviate the problem, *pseudo-label generation*, i.e., assigning pseudo-label to each example by various strategies, has been extensively utilized in previous SPMLL methods. (Cole et al., 2021) initializes all unannotated labels as negative ones and updates the pseudo-labels as learnable parameters with a regularization to constrain the number of expected positive labels. (Xu et al., 2022) employs variational label enhancement (Xu et al., 2023; 2021) to recover latent soft pseudo-labels. (Zhou et al., 2022) adopts asymmetric-tolerance strategies for pseudo-labels cooperating with an entropy-maximization loss. (Xie et al., 2022) recovers the pseudo-labels leveraging the manifold structure information learned by contrastive learning.

By adopting *pseudo-label generation*, existing methods have empirically validated that SPMLL would significantly reduce the amount of supervision with a tolerable damage in classification performance. However, there is no existing method that can provide a theoretical guarantee for learning from pseudo-label on SPMLL. In this paper, the conditions of the effectiveness of learning from pseudo-labels for SPMLL are shown and the learnability of pseudo-label-based methods is proven. Firstly, for any pseudo-label of an instance, it must not always be mislabeled. Secondly, from

[1]School of Computer Science and Engineering, Southeast University, Nanjing 210096, China [2]RIKEN Center for Advanced Intelligence Project, Tokyo 103-0027, Japan. Correspondence to: Ning Xu <xning@seu.edu.cn>, Xin Geng <xgeng@seu.edu.cn>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

a data-generative perspective, for any positive label of an instance, there should be a non-zero probability of being selected as the only single positive label.

Based on the theoretical guarantee of pseudo-label for SPMLL, we propose a novel SPMLL method named Mutual label enhancement for sIngle-positive Multi-label lEarning (MIME). Specifically, label-specific features are learned from the perspective of mutual information for reducing the complexity of original features and preserving their essential characteristics for a certain label. In addition, we prove that the generated pseudo-labels by MIME will approximately converge to the fully-supervised case. The contributions are summarized as follows:

- Theoretically, we for the first time provide the conditions of the effectiveness on pseudo-label for SPMLL and demonstrate the learnability of pseudo-label-based methods.
- Practically, we propose a novel *pseudo-label generation* method named MIME for SPMLL, which is theoretically-guaranteed that the generated pseudo-labels by MIME will approximately converge to the fully-supervised case.

Experiments on four multi-label image classification (MLIC) datasets and five MLL datasets show the effectiveness of our methods over several existing SPMLL approaches.

2. Related Work

Multi-label learning is a type of supervised machine learning technique where an instance can be assigned multiple labels simultaneously. To learn from MLL examples, label correlations have been extensively studied, which can be divided into first-order, second-order, and high-order correlations. First-order focuses on extending binary classification algorithms to multi-label learning, such as treating each label as a separate binary classification problem (Boutell et al., 2004; Read et al., 2011). Second-order models label correlations through pairwise label correlations (Elisseeff & Weston, 2001; Fürnkranz et al., 2008). High-order considers the correlations between multiple labels, such as utilizing graph convolutional neural networks to mine the correlations information between all label nodes (Chen et al., 2019). In addition, there has been a growing interest in the use of label-specific features. Label-specific features are features specifically designed to capture the characteristics of a particular label and improve the performance of the models (Yu & Zhang, 2022; Hang & Zhang, 2022).

In reality, it is intractable to accurately annotate each label of each instance for multi-label learning due to the high volume of the output space. Then multi-label learning with missing labels (MLML) is proposed (Sun et al., 2010). MLML methods are mainly based on low-rank, embedding, and graph-based models. The presence of label correlations suggests that the output space is of low-rank (Liu et al., 2021), which has been widely used to complement the missing entries of a label matrix (Xu et al., 2013; Yu et al., 2014; Xu et al., 2016). Another prevalent approach is to follow the paradigm of embedding techniques that map the label vectors to a low-dimensional space where the features and labels are usually jointly embedded in to explore the complementary between feature space and label space (Yeh et al., 2017; Wang, 2019). Furthermore, the graph-based model is a popular solution for MLML, which constructs a label-specific graph for each label from a feature-induced similarity graph and adds a manifold regularization to the empirical risk minimization framework (Sun et al., 2010; Wu et al., 2014).

As an extreme case of multi-label learning with missing labels, only one of the multiple positive labels can be observed in SPMLL. In the earliest work, all unannotated labels are initialized as negative ones and the pseudo-labels are updated as learnable parameters with a regularization to constrain the number of expected positive labels (Cole et al., 2021). Besides, the latent soft labels are recovered in a label enhancement process to train the multi-label classifier (Xu et al., 2022). Additionally, asymmetric pseudo-label is proposed which adopts asymmetric-tolerance strategies for pseudo-labels cooperating with an entropy-maximization loss (Zhou et al., 2022). Furthermore, (Xie et al., 2022) designs a label-aware global consistency regularization to recover the pseudo-labels leveraging the manifold structure information learned by contrastive learning.

3. Preliminaries

3.1. Multi-Label Learning

Multi-label learning (MLL) aims to train a model on the examples that are associated with multiple labels and obtain a predictive model that is able to predict the relevant labels for an unknown instance accurately. Let $\mathcal{X} = \mathbb{R}^q$ denote the instance space and $\mathcal{Y} = \{0, 1\}^c$ denote the label space with c classes. Given the MLL training set $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)|1 \leq i \leq n\}$ where $\boldsymbol{x}_i \in \mathcal{X}$ is a q -dimensional instance and $\boldsymbol{y}_i \in \mathcal{Y}$ is its corresponding labels. Here, $\boldsymbol{y}_i = [y_i^1, y_i^2, \ldots, y_i^c]$ where $y_i^j = 1$ indicates that the j-th label is a relevant label associated with $\boldsymbol{x}_i$ and $y_i^j = 0$ indicates that the j-th label is irrelevant to $\boldsymbol{x}_i$. Multi-label learning is intended to produce a multi-label classifier in the hypothesis space $h \in \mathcal{H} : \mathcal{X} \mapsto \mathcal{Y}$ that minimizes the following classification risk:

$$\mathcal{R}(h) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim p(\boldsymbol{x},\boldsymbol{y})} \left[\mathcal{L}(h(\boldsymbol{x}), \boldsymbol{y})\right], \tag{1}$$

where $\mathcal{L} : \mathcal{X} \times \mathcal{Y} \mapsto \mathbb{R}^+$ is a multi-label loss function that measures the accuracy of the model in fitting the data.

The hamming loss is a widely used loss function for multi-label learning which concerns how many instance-label pairs are misclassified (Gao & Zhou, 2011; Wu & Zhou, 2017). For a given classifier $h : \mathcal{X} \mapsto \mathcal{Y}$, the hamming loss is given by:

$$\mathcal{R}_{\text{ham}}(h) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim p(\boldsymbol{x},\boldsymbol{y})}[\frac{1}{c}\sum_{j=1}^{c}\mathbf{1}(h^j(\boldsymbol{x}) \neq y^j)], \quad (2)$$

where $\mathbf{1}(\cdot)$ is an indicator function and h^j is the j-th output label of the multi-label classifier.

3.2. Single-Positive Multi-Label Learning

For single-positive multi-label learning (SPMLL), each instance is annotated with only one positive label. Given the SPMLL training set $\tilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \gamma_i)|1 \leq i \leq n\}$ where $\gamma_i \in \{1, 2, \ldots, c\}$ denotes the only observed single positive label of $\boldsymbol{x}_i$. The task of SPMLL is to induce a multi-label classifier $h \in \mathcal{H} : \mathcal{X} \mapsto \mathcal{Y}$ from $\tilde{\mathcal{D}}$, which can assign the unknown instance with a set of relevant labels.

Pseudo-label generation, which aims to assign pseudo-label to each example by various strategies, has been extensively utilized in previous SPMLL methods. Let $\boldsymbol{l} = [l^1, l^2, \ldots, l^c]$ denote the pseudo-labels generated by some methods where $l^j = 1$ indicates that j-th label is a relevant label and vice versa. Here, l^γ is usually fixed as 1 where γ is the label index of the only single positive label of $\boldsymbol{x}$. Add the generated pseudo-labels to the dataset $\tilde{\mathcal{D}}$, a dataset with pseudo-labels $\bar{\mathcal{D}} = \{(\boldsymbol{x}_i, \gamma_i, \boldsymbol{l}_i)|1 \leq i \leq n\}$ is obtained. For a given classifier $h : \mathcal{X} \mapsto \mathcal{Y}$, the hamming loss of the classifier trained by the SPMLL training set with pseudo-labels $\bar{\mathcal{D}}$ is given by:

$$\hat{\mathcal{R}}_{\text{ham}}(h) = \frac{1}{nc}\sum_{i=1}^{n}\sum_{j=1}^{c}\mathbf{1}(h^j(\boldsymbol{x}_i) \neq l_i^j). \quad (3)$$

An Empirical Risk Minimizing (ERM) learner $\mathcal{A}$ for $\mathcal{H}$ is a function $\mathcal{A} : \cup_{n=0}^{\infty}(\mathcal{X} \times \mathcal{Y})^n \mapsto \mathcal{H}$. The ERM learner for hypothesis space $\mathcal{H}$ returns a hypothesis $h \in \mathcal{H}$ with minimizing the hamming loss of the classifier trained by pseudo-labels on the SPMLL training set $\bar{\mathcal{D}}$.

$$\mathcal{A}(\bar{\mathcal{D}}) = \arg\min_{h\in\mathcal{H}} \hat{\mathcal{R}}_{\text{ham}}(h). \quad (4)$$

It is noticed that in the implementation, the hamming loss is often replaced by binary cross entropy loss (BCE) for convenience of derivation in previous SPMLL methods (Cole et al., 2021; Xu et al., 2022; Xie et al., 2022).

4. Pseudo-Label Generation for SPMLL

Existing methods have empirically demonstrated that SPMLL can reduce supervision with a tolerable reduction in classification performance by utilizing *pseudo-label generation*. Nevertheless, no existing method is able to provide a theoretical guarantee for pseudo-label on SPMLL. In this section, two conditions under which pseudo-label-based methods is effective for SPMLL is discussed. Firstly, any pseudo-label of an instance should not always be misannotated; In addition, any positive label assigned to an instance should have a non-zero probability of being selected as the only positive label.

4.1. Small Unreliability Degree Condition

We define the *unreliability degree* of pseudo-labels as:

$$\xi = \sup_{\substack{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{l})\sim p(\boldsymbol{x},\boldsymbol{y},\boldsymbol{l}),\\ j\in\{1,2,\ldots,c\}}} \Pr(l^j \neq y^j). \quad (5)$$

The *unreliability degree* describes how much the pseudo-labels generated by some pseudo-label-based methods are different from the ground-truth labels. If $\xi = 0$, then with probability one there are no mislabeled pseudo-labels. Intuitively, a model trained with pseudo-labels is more effective if the generated pseudo-labels method is more accurate. The following is a theoretical explanation of this intuition.

Theorem 4.1. *Suppose a SPMLL pseudo-label-based method has an unreliability degree of pseudo-label ξ, $0 \leq \xi < 1$, Let $\theta_1 = c\log\frac{2}{1+\xi}$, and suppose the Natarajan dimension of the hypothesis space $\mathcal{H}$ is $d_{\mathcal{H}}$, define*

$$n_0(\mathcal{H}, \epsilon, \delta) = \frac{4}{\theta_1\epsilon}\left(d_{\mathcal{H}}\left(\log(4d_{\mathcal{H}}) + 2c\log c + \log\frac{1}{\theta_1\epsilon}\right) + \log\frac{1}{\delta} + 1\right).$$

Then when $n > n_0(\mathcal{H}, \epsilon, \delta)$, $\mathcal{R}_{ham}(\mathcal{A}(\bar{\mathcal{D}})) < \epsilon$ with probability $1 - \delta$.

The proof is provided in Appendix A.1. The *unreliability degree* of the generated pseudo-labels should satisfy the *small unreliability degree condition*, i.e., $\xi < 1$. If $\xi = 1$, there exists at least one pseudo-label that always differs from its ground-truth label. Then the always mislabeled pseudo-label is unlearnable for the ERM learner.

4.2. Non-Zero Minimum Positive Label Sampling Probability Condition

The small unreliability degree is a condition focusing on the pseudo-label methods. In this section, we will give a condition from a data-generative perspective and demonstrate the learnability of SPMLL. The *minimum positive*

label sampling probability is defined as:

$$\tau = \inf_{\substack{(\boldsymbol{x},\boldsymbol{y},\gamma)\sim p(\boldsymbol{x},\boldsymbol{y},\gamma),\\ \boldsymbol{y}_j=1, j\in\{1,2,\ldots,c\}}} \Pr\left(j=\gamma\right). \tag{6}$$

The *minimum positive label sampling probability* describes the minimum probability of the positive labels to be sampled as the only single positive label of an instance.

Theorem 4.2. *Suppose a SPMLL problem has $\tau > 0$, Let $\theta_2 = c\log\frac{2}{2-\tau}$, and suppose the Natarajan dimension of the hypothesis space $\mathcal{H}$ is $d_{\mathcal{H}}$, define*

$$n_0(\mathcal{H},\epsilon,\delta) = \frac{4}{\theta_2\epsilon}\left(d_{\mathcal{H}}\left(\log\left(4d_{\mathcal{H}}\right) + 2c\log c + \log\frac{1}{\theta_2\epsilon}\right) + \log\frac{1}{\delta} + 1\right).$$

Then when $n > n_0(\mathcal{H},\epsilon,\delta)$, $\mathcal{R}(\mathcal{A}(\bar{\mathcal{D}})) < \epsilon$ with probability $1-\delta$.

The proof is provided in Appendix A.2. Under the condition that each positive label of each instance is capable of being sampled as the single positive label ($\tau > 0$), although the pseudo-labels are randomly labeled, i.e., $\xi = 1$, the ERM learner can still return an ideal hypothesis. If $\tau = 0$, there exists at least one positive label that is not possible to be sampled as the single positive label. Then this label is unlearnable for the ERM learner.

5. The MIME Approach

The MIME approach designs a target function from the perspective of mutual information, which can simultaneously train the model and update the pseudo-labels in a label enhancement process (Xu et al., 2023; 2021). Specifically, label-specific features (Yu & Zhang, 2022) are induced by the information bottleneck (Tishby et al., 2000; Alemi et al., 2017) principle, reducing the complexity of original features while preserving its essential characteristics for a certain label. Then the learned label-specific features can further assist in improving the prediction of the model and estimating the mutual information. Subsequently, the pseudo-labels can be updated more precisely based on the estimated mutual information.

Consider $\boldsymbol{x}$ and y^j as random variables of the original features and j-th label respectively, and let $\boldsymbol{z}^j$ be the extracted label-specific features of j-th label. Information bottleneck expresses the tradeoff between the mutual information measures $I(\boldsymbol{x},\boldsymbol{z}^j)$ and $I(\boldsymbol{z}^j,y^j)$, where $I(\boldsymbol{x},\boldsymbol{z}^j)$ and $I(\boldsymbol{z}^j,y^j)$ respectively quantify the amount of information that the label-specific feature contains about the original features and j-th label. Then we can maximize the objective function:

$$\mathcal{L}_{IB} = \sum_{j=1}^{c} I(\boldsymbol{z}^j,y^j) - \beta_j I(\boldsymbol{z}^j,\boldsymbol{x}). \tag{7}$$

Here the goal is to learn encoding $\boldsymbol{z}^j$ that is maximally expressive about y^j while being maximally compressive about $\boldsymbol{x}$, where $\beta_j \geq 0$ controls the tradeoff. The first term in Eq. (7) encourages $\boldsymbol{z}^j$ to be indicative of y^j and the second term encourages $\boldsymbol{z}^j$ to discard the redundant information of $\boldsymbol{x}$.

However, for SPMLL, the original accurate supervision label y^j is unavailable for Eq. (7). Let l^j be the random variable of j-th pseudo-labels. A lower bound is obtained for Eq. (7) under a mild assumption that $H(l^j|\boldsymbol{z}^j) \geq H(y^j|\boldsymbol{z}^j)$ [1]:

$$\begin{aligned}\mathcal{L}_{IB} &= \sum_{j=1}^{c} I(\boldsymbol{z}^j,y^j) - \beta_j I(\boldsymbol{z}^j,\boldsymbol{x})\\ &\geq \sum_{j=1}^{c} I(\boldsymbol{z}^j,l^j) - \beta_j I(\boldsymbol{z}^j,\boldsymbol{x}).\end{aligned} \tag{8}$$

Information entropy $H(\cdot)$ is a measure of the uncertainty of a random variable. Information bottleneck believes that extracted features $\boldsymbol{z}^j$ contains most of the information that can be used to predict y^j while the pseudo-labels l^j is not only related to $\boldsymbol{z}^j$, but also to other factors (such as the methods to generate the pseudo-labels). Therefore, the uncertainty of l^j is larger than that of y^j when $\boldsymbol{z}^j$ is known, thus we make the assumption that $H(l^j|\boldsymbol{z}^j) \geq H(y^j|\boldsymbol{z}^j)$. The next step is optimizing the objective function:

$$\mathcal{L} = \sum_{j=1}^{c} I(\boldsymbol{z}^j,l^j) - \beta_j I(\boldsymbol{z}^j,\boldsymbol{x}). \tag{9}$$

We use variational inference to construct a lower bound for Eq. (9) as the approximate methods proposed in (Alemi et al., 2017).

The joint distribution $p(\boldsymbol{x},\boldsymbol{z}^j,l^j)$ is decomposed as:

$$\begin{aligned}p(\boldsymbol{x},\boldsymbol{z}^j,l^j) &= p(\boldsymbol{z}^j|\boldsymbol{x},l^j)p(l^j|\boldsymbol{x})p(\boldsymbol{x})\\ &= p(\boldsymbol{z}^j|\boldsymbol{x})p(l^j|\boldsymbol{x})p(\boldsymbol{x}),\end{aligned} \tag{10}$$

where we assume $p(\boldsymbol{z}^j|\boldsymbol{x},l^j) = p(\boldsymbol{z}^j|\boldsymbol{x})$, this restriction means that the label-specific features $\boldsymbol{z}^j$ do not depend directly on the pseudo-labels l^j and it only relys on the original features $\boldsymbol{x}$. This assumption promotes the appearance of unsupervised learning (Saunshi et al., 2019; He et al., 2020). The first term of Eq. (9) can be written in full as:

$$\begin{aligned}I(\boldsymbol{z}^j,l^j) &= \int p(l^j,\boldsymbol{z}^j)\log\frac{p(l^j,\boldsymbol{z}^j)}{p(l^j)p(\boldsymbol{z}^j)}dl^j d\boldsymbol{z}^j\\ &= \int p(l^j,\boldsymbol{z}^j)\log\frac{p(l^j|\boldsymbol{z}^j)}{p(l^j)}dl^j d\boldsymbol{z}^j.\end{aligned} \tag{11}$$

[1] The detail is provided in Appendix A.3.

Algorithm 1 MIME Algorithm

Input: The SPMLL training set $\tilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \gamma_i)|1 \leq i \leq n\}$, a threshold τ, the number of iteration I and the number of epoch T.

Initialize the pseudo-labels with AN solution (assuming unannotated labels as negative ones) and get the initialized training set with pseudo-labels $\bar{\mathcal{D}} = \{(\boldsymbol{x}_i, \boldsymbol{l}_i)|1 \leq i \leq n\}$.
Warm up the model with Eq. (20) and initialize the model with parameters $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$.
for $t = 1$ **to** T **do**
  for $k = 1$ **to** I **do**
    Fetch a random mini-batch $\mathcal{B}$ from $\bar{\mathcal{D}}$;
    Update the parameters $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ with Eq. (20).
  end for
  for $i = 1$ **to** n **do**
    For each instance and its pseudo-label $(\boldsymbol{x}_i, \boldsymbol{l}_i)$.
    for $j = 1$ **to** c **do**
      Add the j-th label into the positive label set and get a new pseudo-label vector $\boldsymbol{l}_i^{\text{new}}$;
      if $\ell(\boldsymbol{x}_i, \boldsymbol{l}_i^{\text{new}}) - \ell(\boldsymbol{x}, \boldsymbol{l}_i) \geq \tau$ **then**
        Add the j-th label into the positive label set and update the pseudo-label vector of $\boldsymbol{x}_i$ as $\boldsymbol{l}_i^{\text{new}}$.
      end if
    end for
  end for
end for

Output: The parameters of model $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$.

Since $p(l^j|\boldsymbol{z}^j)$ is intractable, $q(l^j|\boldsymbol{z}^j)$ is employed to approximate $p(l^j|\boldsymbol{z}^j)$. Based on the fact that the Kullback-Leibler divergence is always positive:

$$\begin{aligned}&\mathrm{KL}\left[p(l^j|\boldsymbol{z}^j)\|q(l^j|\boldsymbol{z}^j)\right] = \\ &-\int p(l^j|\boldsymbol{z}^j)\log\frac{q(l^j|\boldsymbol{z}^j)}{p(l^j|\boldsymbol{z}^j)}dl^j d\boldsymbol{z}^j \geq 0.\end{aligned} \tag{12}$$

we have:

$$\int p(l^j|\boldsymbol{z}^j)\log p(l^j|\boldsymbol{z}^j)dl^j \geq \int p(l^j|\boldsymbol{z}^j)\log q(l^j|\boldsymbol{z}^j)dl^j, \tag{13}$$

then:

$$\begin{aligned}I(\boldsymbol{z}^j, l^j) &\geq \int p(l^j, \boldsymbol{z}^j)\log\frac{q(l^j|\boldsymbol{z}^j)}{p(l^j)}dl^j d\boldsymbol{z}^j \\ &= \int p(l^j, \boldsymbol{z}^j)\log q(l^j|\boldsymbol{z}^j)dl^j d\boldsymbol{z}^j + H(l^j) \\ &\geq \int p(l^j, \boldsymbol{z}^j)\log q(l^j|\boldsymbol{z}^j)dl^j d\boldsymbol{z}^j.\end{aligned} \tag{14}$$

For Eq. (14), $p(l^j, \boldsymbol{z}^j)$ can be rewritten as $p(l^j, \boldsymbol{z}^j) = \int p(\boldsymbol{x}, l^j, \boldsymbol{z}^j)d\boldsymbol{x} = \int p(\boldsymbol{x})p(l^j|\boldsymbol{x})p(\boldsymbol{z}^j|\boldsymbol{x})d\boldsymbol{x}$ according to the assumption $p(\boldsymbol{z}^j|\boldsymbol{x}, l^j) = p(\boldsymbol{z}^j|\boldsymbol{x})$ in Eq. (10). Then a new lower bound is given:

$$\begin{aligned}I(\boldsymbol{z}^j, l^j) \geq \int &p(\boldsymbol{x})p(l^j|\boldsymbol{x})p(\boldsymbol{z}^j|\boldsymbol{x}) \\ &\log q(l^j|\boldsymbol{z}^j)dl^j d\boldsymbol{z}^j d\boldsymbol{x}.\end{aligned} \tag{15}$$

We now consider the second term $I(\boldsymbol{z}^j, \boldsymbol{x})$ in Eq. (9):

$$I(\boldsymbol{z}^j, \boldsymbol{x}) = \int p(\boldsymbol{x}, \boldsymbol{z}^j)\log\frac{p(\boldsymbol{z}^j|\boldsymbol{x})}{p(\boldsymbol{z}^j)}d\boldsymbol{z}^j d\boldsymbol{x}. \tag{16}$$

Due to the difficulty of computing the marginal distribution $p(\boldsymbol{z}^j) = \int p(\boldsymbol{z}^j|\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x}$, let $r(\boldsymbol{z}^j)$ be a variational approximation of $p(\boldsymbol{z}^j)$. Similar to Eq. (12), since $\mathrm{KL}[p(\boldsymbol{z}^j)0\|r(\boldsymbol{z}^j)] \geq 0 \Longrightarrow \int p(\boldsymbol{z}^j)\log p(\boldsymbol{z}^j)d\boldsymbol{z}^j \geq \int p(\boldsymbol{z}^j)\log r(\boldsymbol{z}^j)d\boldsymbol{z}^j$, then we have:

$$I(\boldsymbol{z}^j, \boldsymbol{x}) \leq \int p(\boldsymbol{x})p(\boldsymbol{z}^j|\boldsymbol{x})\log\frac{p(\boldsymbol{z}^j|\boldsymbol{x})}{r(\boldsymbol{z}^j)}d\boldsymbol{x}d\boldsymbol{z}^j. \tag{17}$$

According to Eq. (15) and Eq. (17), the objective function (9) can be bounded by:

$$\begin{aligned}&I(\boldsymbol{z}^j, l^j) - \beta_j I(\boldsymbol{z}^j, \boldsymbol{x}) \geq \\ &\int p(\boldsymbol{x})p(l^j|\boldsymbol{x})p(\boldsymbol{z}^j|\boldsymbol{x})\log q(l^j|\boldsymbol{z}^j)dl^j d\boldsymbol{z}^j d\boldsymbol{x} \\ &-\beta_j\int p(\boldsymbol{x})p(\boldsymbol{z}^j|\boldsymbol{x})\log\frac{p(\boldsymbol{z}^j|\boldsymbol{x})}{r(\boldsymbol{z}^j)}d\boldsymbol{x}d\boldsymbol{z}^j.\end{aligned} \tag{18}$$

The empirical data distribution $p(\boldsymbol{x}, \boldsymbol{l}) = \frac{1}{n}\sum_{i=1}^{n}\delta_{\boldsymbol{x}_i}(\boldsymbol{x})\Pi_{j=1}^{c}\delta_{l_i^j}(l^j)$ is employed to approximate the joint distribution $p(\boldsymbol{x}, \boldsymbol{l}) = p(\boldsymbol{x})p(\boldsymbol{l}|\boldsymbol{x})$, then we have:

$$\begin{aligned}\mathcal{L} &= \sum_{j=1}^{c} I(\boldsymbol{z}^j, l^j) - \beta_j I(\boldsymbol{z}^j, \boldsymbol{x}) \\ &\approx \frac{1}{nc}\sum_{i=1}^{n}\sum_{j=1}^{c}\Bigg[\int p(\boldsymbol{z}^j|\boldsymbol{x}_i)\log q(l_i^j|\boldsymbol{z}^j) \\ &\quad -\beta_j p(\boldsymbol{z}^j|\boldsymbol{x}_i)\log\frac{p(\boldsymbol{z}^j|\boldsymbol{x}_i)}{r(\boldsymbol{z}^j)}d\boldsymbol{z}^j\Bigg],\end{aligned} \tag{19}$$

where the label-specific feature of j-th label $\boldsymbol{z}^j$ follows a Gaussion distribution $p(\boldsymbol{z}^j|\boldsymbol{x}) = \mathcal{N}\left(\boldsymbol{z}^j|f_e^{\mu}(\boldsymbol{x}), f_e^{\Sigma}(\boldsymbol{x})\right)$ encoded by an inference model f_e that outputs both the k-dimensional mean of $\boldsymbol{z}^j$ and the $k \times k$ covariance matrix Σ. Note that the implicit reparameterization gradient (Figurnov et al., 2018) is employed to write $p(\boldsymbol{z}^j|\boldsymbol{x})d\boldsymbol{z}^j = p(\epsilon)d\epsilon$, which avoids the inversion of the standardization function and makes the gradients can be computed analytically in backward pass, where $\boldsymbol{z}^j = f(\boldsymbol{x}, \epsilon)$ is a deterministic function of $\boldsymbol{x}$ and the Gaussian random variable ϵ.

Assuming our selection of the posterior probability distribution $p(\boldsymbol{z}^j|\boldsymbol{x})$ and the prior probability distribution $r(\boldsymbol{z}^j)$

facilitates the computation of an analytical Kullback-Leibler divergence. Then the objective function Eq. (19) is rewritten as:

$$\mathcal{L} \approx \frac{1}{nc}\sum_{i=1}^{n}\sum_{j=1}^{c}\mathbb{E}_{\epsilon\sim p(\epsilon)}\left[-\log q_{\boldsymbol{\phi}}\left(l_i^j|f_{\boldsymbol{\theta}}\left(\boldsymbol{x}_i,\epsilon\right)\right)\right] + \beta_j \,\mathrm{KL}\left[p\left(\boldsymbol{z}^j|\boldsymbol{x}_i\right)\|r\left(\boldsymbol{z}^j\right)\right], \tag{20}$$

where $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$ is the parameters of the decoder model q and encoder model f respectively. We assume that the prior density $r(\boldsymbol{z}^j)$ is the product of standard Gaussian and $p(\boldsymbol{z}^j|\boldsymbol{x}_i)$ is the product of Gaussian parameterized by the mean vector $\boldsymbol{\mu}^j = [\mu_1^j, \ldots, \mu_k^j]$ and standard deviation vector $\boldsymbol{\sigma}^j = [\sigma_1^j, \ldots, \sigma_k^j]$. Then:

$$\mathrm{KL}[p(\boldsymbol{z}^j|\boldsymbol{x}_i)\|r(\boldsymbol{z}^j)] = -\frac{1}{2}\sum_{i=1}^{k}\Big(1+2\log((\sigma_i^j)) - (\mu_i^j)^2 - (\sigma_i^j)^2\Big).$$

We now discuss how to update the pseudo-labels $\boldsymbol{l}$ of instance $\boldsymbol{x}$. Recall the original objective function (9), the second term of the equation is irrelevant to the pseudo-labels and can be ignored; the first term can be expanded by the empirical data distribution $p(\boldsymbol{x},\boldsymbol{l}) = \frac{1}{n}\sum_{i=1}^{n}\delta_{\boldsymbol{x}_i}(\boldsymbol{x})\delta_{l_i^j}(\boldsymbol{l})$ as:

$$\begin{aligned}\sum_{j=1}^{c} I(\boldsymbol{z}^j, l^j) &= \sum_{j=1}^{c}\int p(\boldsymbol{x})p(l^j|\boldsymbol{x})p(\boldsymbol{z}^j|\boldsymbol{x}) \\ &\qquad \log\frac{p(l^j,\boldsymbol{z}^j)}{p(l^j)p(\boldsymbol{z}^j)}dl^j d\boldsymbol{z}^j d\boldsymbol{x} \\ &= \sum_{i=1}^{n}\sum_{j=1}^{c}\int p(\boldsymbol{z}^j|\boldsymbol{x}_i)\log\frac{p(l_i^j|\boldsymbol{z}^j)}{p(l_i^j)}d\boldsymbol{z}^j.\end{aligned} \tag{21}$$

For convenience, we only consider a single instance and define a score function:

$$\begin{aligned}\ell(\boldsymbol{x},\boldsymbol{l}) &= \sum_{j=1}^{c}\int p(\boldsymbol{z}^j|\boldsymbol{x})\log\frac{p(l^j|\boldsymbol{z}^j)}{p(l^j)}d\boldsymbol{z}^j \\ &= \sum_{j=1}^{c}\mathbb{E}_{\boldsymbol{z}^j\sim p(\boldsymbol{z}^j|\boldsymbol{x})}\left[\log p(l^j|\boldsymbol{z}^j) - \log p(l^j)\right]\end{aligned} \tag{22}$$

Intuitively, the score of a function increases when a a true positive label is added to the identified positive label set comparing with a false positive label. The unannotated positive labels can be identified by utilizing the property that the score function has a larger value if the identified positive label set is more accurate. Then, we define the ϵ-identifiable score function as:

Definition 5.1. (ε-identifiable Score Function) Let s be the identified label set where the labels in the set are all positive labels and let y be a positive label that is not in the identified label set. A score function $g : \mathcal{X}\times 2^{[c]}\mapsto\mathbb{R}^+$ is called ϵ-identifiable Score Function if it have $g(\boldsymbol{x}, s\cup\{y\}) - g(\boldsymbol{x}, s) \geq \varepsilon$.

The following theorem is derived from this definition, which demonstrates that Eq. (22) is a ϵ-identifiable Score Function and it is capable of identifying the true positive label.

Theorem 5.2. *$g(\boldsymbol{x}, s) = \ell(\boldsymbol{x}, \boldsymbol{l}_s)$ is a ε-identifiable Score Function for all $0 < \varepsilon \leq \min_{1\leq j\leq c}\mathbb{E}_{\boldsymbol{z}^j\sim p(\boldsymbol{z}^j|\boldsymbol{x})}\left[\log\frac{p(l^j=1|\boldsymbol{z}^j)}{p(l^j=0|\boldsymbol{z}^j)}\right]$, where the j-th item of $\boldsymbol{l}_s$ is 1 if j-th label is in the identified label set s and vice versa.*

The proof is provided in Appendix A.4. Initializing the identified positive label set by adding the only positive label to it, the other positive labels can be identified if the score function has a larger value when a new label is added in the identified positive label set.

MIME first initializes the pseudo-labels by assuming unannotated labels as negative ones and warm-up the model to attain a fine network before it starts fitting noise (Zhang et al., 2017). Then the steps of optimizing Eq. (20) and enhancing the pseudo-labels are iterated alternately. The algorithmic description is shown in Algorithm 1.

6. Experiments

6.1. Experimental Configurations

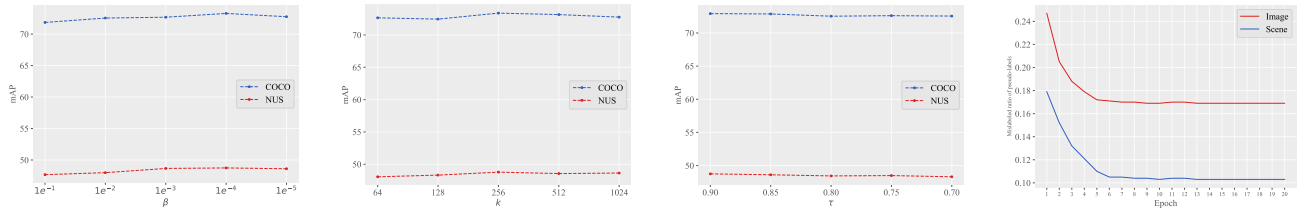
Datasets: In the experiments, following (Cole et al., 2021; Xu et al., 2022), we employed four large scale multi-label image classification (MLIC) datasets and five widely-used MLL datasets (Hang & Zhang, 2022) to evaluate our proposed method. The four MLIC datasets include `PSACAL VOC 2021 (VOC)` (Everingham et al., 2010), `MS-COCO 2014 (COCO)` (Lin et al., 2014), `NUS-WIDE (NUS)` (Chua et al., 2009), and `CUB-200 2011 (CUB)` (Wah et al., 2011); the five MLL datasets cover a wide range of scenarios with heterogeneous multi-label characteristics. For each MLIC dataset, we withhold 20% of the training set for validation. For each MLL dataset, we split the dataset as train/validation/test set in a ratio of 80%/10%10%. One of the positive labels is randomly selected for each training instance, while the validation and test sets remain fully labeled. The details of these datasets are provided in Appendix A.6. Following the experimental setting in previous SPMLL literature, *Mean average precision (mAP)* is employed on the four MLIC datasets (Cole et al., 2021; Xie et al., 2022; Zhou et al., 2022) and five popular multi-label metrics are employed on the MLL datasets including *Ranking loss, Hamming loss, One-error, Coverage*, and *Average precision* (Xu et al., 2022).

Table 1: Predictive performance of each comparing methods on four MLIC datasets in terms of *mean average precision (mAP)* (mean ± std). The best performance is highlighted in bold (the larger the better).

	VOC	COCO	NUS	CUB
AN	85.546±0.294	64.326±0.204	42.494±0.338	18.656±0.090
AN-LS	87.548±0.137	67.074±0.196	43.616±0.342	16.446±0.269
WAN	87.138±0.240	65.552±0.171	45.785±0.192	14.622±1.300
EPR	85.228±0.444	63.604±0.249	45.240±0.338	19.842±0.423
ROLE	88.088±0.167	67.022±0.141	41.949±0.205	14.798±0.613
EM	88.674±0.077	70.636±0.094	47.254±0.297	20.692±0.527
EM-APL	88.860±0.080	70.758±0.215	47.778±0.181	21.202±0.792
SMILE	86.311±0.450	63.331±0.112	43.611±0.172	18.611±0.144
LAGC	88.021±0.121	70.422±0.062	46.211±0.155	21.840±0.237
MIME	**89.199±0.157**	**72.920±0.255**	**48.743±0.428**	**21.890±0.347**

Table 2: Predictive performance of each comparing methods on MLL datasets in terms of *Average precision (AP)* (mean ± std). The best performance is highlighted in bold (the larger the better).

	Image	Scene	Yeast	Rcv1subset1	Mediamill
AN	0.599±0.029	0.694±0.095	0.719±0.006	0.501±0.002	0.690±0.002
AN-LS	0.668±0.022	0.737±0.043	0.735±0.002	0.548±0.002	0.696±0.001
WAN	0.672±0.031	0.765±0.053	0.730±0.003	0.551±0.002	0.687±0.002
EPR	0.658±0.020	0.713±0.037	0.729±0.002	0.502±0.003	0.606±0.017
ROLE	0.625±0.045	0.752±0.049	0.740±0.004	0.561±0.003	0.690±0.005
EM	0.531±0.039	0.724±0.032	0.682±0.093	0.582±0.003	0.691±0.002
EM-APL	0.521±0.011	0.622±0.069	0.718±0.012	0.563±0.002	0.686±0.001
SMILE	0.755±0.032	0.801±0.060	0.721±0.002	0.581±0.001	0.687±0.002
MIME	**0.777±0.028**	**0.824±0.035**	**0.753±0.002**	**0.594±0.009**	**0.694±0.006**



(a) Sensitivity analysis of β. (b) Sensitivity analysis of k. (c) Sensitivity analysis of τ. (d) The mislabeled ratio curves.

Figure 1: (a) The sensitivity analysis of tradeoff parameters β. (b) The sensitivity analysis of dimension of label-specific features k. (c) The sensitivity analysis of the threshold τ. (d) The mislabeled ratio curves of pseudo-labels on `Image`.

Comparing methods: Our method is compared with the following: 1) AN (Cole et al., 2021) assumes that the unannotated labels are negative and uses binary cross entropy loss for training. 2) AN-LS (Cole et al., 2021) assumes that the unannotated labels are negative and reduces the impact of the false negative labels by label smoothing. 3) WAN (Cole et al., 2021), in which a weight parameter is introduced in order to down-weight losses in relation to negative labels. 4) EPR (Cole et al., 2021) utilizes a regularization to constrain the number of predicted positive labels. 5) ROLE (Cole et al., 2021) online estimates the unannotated labels as learnable parameters throughout training based on EPR. 6) SMILE (Xu et al., 2022), in which the latent soft labels are recovered in a label enhancement process to train the multi-label classifier with binary cross entropy loss. 7) EM (Zhou et al., 2022) reduces the effect of the incorrect labels by the entropy-maximization loss. 8) EM-APL (Zhou et al., 2022) adopts asymmetric-tolerance pseudo-label strategies cooperating with entropy-maximization loss and then more precise supervision can be provided. 9) LAGC (Xie et al.,

Table 3: Predictive performance of each comparing methods on MLL datasets in terms of *Ranking loss* (mean ± std). The best performance is highlighted in bold (the smaller the better).

	Image	Scene	Yeast	Rcv1subset1	Mediamill
AN	0.336±0.051	0.204±0.084	0.197±0.005	0.099±0.001	0.055±0.000
AN-LS	0.269±0.036	0.155±0.031	0.182±0.002	0.066±0.001	0.059±0.001
WAN	0.275±0.034	0.132±0.028	0.184±0.002	0.065±0.001	0.054±0.000
EPR	0.276±0.026	0.166±0.026	0.186±0.002	0.087±0.002	0.059±0.003
ROLE	0.304±0.048	0.145±0.041	0.179±0.005	0.070±0.003	0.067±0.002
EM	0.425±0.035	0.149±0.018	0.246±0.120	0.051±0.000	0.055±0.001
EM-APL	0.431±0.031	0.220±0.051	0.191±0.005	0.059±0.001	0.055±0.001
SMILE	0.194±0.025	0.124±0.054	0.185±0.005	0.054±0.001	0.055±0.002
MIME	**0.170±0.045**	**0.099±0.023**	**0.171±0.009**	**0.050±0.002**	**0.053±0.006**

Table 4: Predictive performance of each comparing methods on MLL datasets in terms of *One-error* (mean ± std). The best performance is highlighted in bold (the smaller the better).

	Image	Scene	Yeast	Rcv1subset1	Mediamill
AN	0.629±0.042	0.478±0.126	0.240±0.002	0.510±0.006	0.161±0.006
AN-LS	0.526±0.014	0.439±0.068	0.242±0.010	0.498±0.006	0.151±0.009
WAN	0.513±0.041	0.399±0.097	0.239±0.002	0.485±0.008	0.169±0.006
EPR	0.543±0.024	0.482±0.058	0.240±0.000	0.531±0.004	0.486±0.146
ROLE	0.596±0.066	0.409±0.066	0.237±0.010	0.480±0.007	0.167±0.028
EM	0.708±0.073	0.479±0.055	0.240±0.000	0.481±0.006	0.150±0.005
EM-APL	0.721±0.029	0.638±0.105	0.240±0.007	0.481±0.010	0.156±0.002
SMILE	0.374±0.025	0.352±0.101	0.236±0.006	0.468±0.009	0.157±0.005
MIME	**0.360±0.003**	**0.299±0.013**	**0.223±0.004**	**0.440±0.006**	**0.149±0.009**

2022) designs a label-aware global consistency regularization to recover the pseudo-labels leveraging the manifold structure information learned by contrastive learning. The implementation details in experiments is provided in Appendix A.5.

6.2. Experimental Results

Table 1 reports the comparison results on `PSACAL VOC 2021 (VOC)`, `MS-COCO 2014 (COCO)`, `NUS-WIDE (NUS)`, and `CUB-200 2011 (CUB)` in terms of *mAP*. Due to that the adjacency matrix used by SMILE is difficult to obtain for the large-scale MLIC datasets, we use the confidence outputed by the model as the soft label of the unbiased risk estimator in the experiment. As shown in Table 1, our approach consistently surpasses all comparative methods.

Table 2, 3 and 4 report the results of our method and other comparing methods on five MLL datasets in terms of *Average precision, Coverage*, and *One-error* respectively. The results on other metrics are comparable and can be observed in Appendix A.7. It is noticeable that data augmentation techniques used in LAGC cannot be directly applied to the MLL datasets where the input of each instance is a feature vector, so we do not compare LAGC with other methods on MLL datasets. The results demonstrate that our method achieves superior performance compared to all the comparing approaches on the majority of evaluation metrics (except the results of `Yeast` and `Rcv1subset1` on the metrics *Hamming loss* and *Coverage* where our method attains a comparable performance against SMILE).

All of the results validate that our proposed method is capable of effectively addressing SPMLL problem.

6.3. Further analysis

We investigate the effect of the tradeoff parameters β, the dimension of label-specific features k and the threshold τ. The performance curves of MIME on dataset `COCO` and `NUS` are illustrated in Figure 1a, Figure 1b and Figure 1c. The hyperparameters β, k and τ are changed in the range of $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}\}$, $\{64, 128, 256, 512, 1024\}$ and $\{0.9, 0.85, 0.8, 0.75, 0.7\}$ respectively. As shown in the figures, the largest performance gap with respect to the tradeoff parameters β and the dimension of label-specific features k are about 1.9% and

1.2%. The performance curves show that our method is robust against the choice of tradeoff parameters β and the dimension of label-specific features k in a wide range.

Figure 1d illustrates the mislabeled ratio of the generated pseudo-labels on `Image` and `Scene`. As shown in the figure, the mislabeled ratio is steadily decreasing and converges eventually as the epoch increases, which shows that the unannotated positive labels are continously identified by MIME and it is an effective *pseudo-label generation* method for SPMLL.

7. Conclusion

In this paper, we study single-positive multi-label learning and provide the conditions of the effectiveness on pseudo-label for SPMLL and demonstrate the learnability of pseudo-label-based methods. Furthermore, based on the theoretical guarantee of pseudo-label for SPMLL, we propose a novel *pseudo-label generation* method for SPMLL named MIME from the perspective of mutual information. Experiments on four MLIC datasets and five MLL datasets demonstrate the efficacy of our method over several existing SPMLL approaches.

8. Acknowledgments

This research was supported by the National Key Research & Development Plan of China (2018AAA0100104), the National Science Foundation of China (62206050, 62125602, and 62076063), China Postdoctoral Science Foundation (2021M700023), Jiangsu Province Science Foundation for Youths (BK20210220), Young Elite Scientists Sponsorship Program of Jiangsu Association for Science and Technology (TJ-2022-078), and the Big Data Computing Center of Southeast University.

References

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. Deep variational information bottleneck. In *Proceedings of 5th International Conference on Learning Representations*, Toulon, France, 2017.

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

Chen, S., Wang, J., Chen, Y., Shi, Z., Geng, X., and Rui, Y. Label distribution learning on auxiliary label space graphs for facial expression recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13981–13990, Seattle, WA, 2020.

Chen, Z., Wei, X., Wang, P., and Guo, Y. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5177–5186, Long Beach, CA, 2019.

Chua, T., Tang, J., Hong, R., Li, H., Luo, Z., and Zheng, Y. NUS-WIDE: a real-world web image database from national university of singapore. In *Proceedings of the 8th ACM International Conference on Image and Video Retrieval*, Santorini Island, Greece, 2009.

Cole, E., Aodha, O. M., Lorieul, T., Perona, P., Morris, D., and Jojic, N. Multi-label learning from single positive labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 933–942, virtual, 2021.

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

Everingham, M., Gool, L. V., Williams, C. K. I., Winn, J. M., and Zisserman, A. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

Figurnov, M., Mohamed, S., and Mnih, A. Implicit reparameterization gradients. In *Advances in Neural Information Processing Systems 31*, pp. 439–450, Montréal, Canada, 2018.

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

Gao, W. and Zhou, Z. On the consistency of multi-label learning. In *Proceedings of the 24th annual conference on learning theory*, volume 19, pp. 341–358, Budapest, Hungary, 2011.

Hang, J. and Zhang, M. Collaborative learning of label semantics and deep label-specific features for multi-label classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9860–9871, 2022.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, 2016.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. B. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9726–9735, Seattle, WA, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International*

Conference on Learning Representations, San Diego, CA, 2015.

Lin, T., Maire, M., Belongie, S. J., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft COCO: common objects in context. In *Proceedings of 13th European Conference on Computer Vision*, volume 8693, pp. 740–755, Zurich, Switzerland, 2014.

Liu, J., Chang, W., Wu, Y., and Yang, Y. Deep learning for extreme multi-label text classification. In *Proceedings of the 40-th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 115–124, Tokyo, Japan, 2017.

Liu, L. and Dietterich, T. G. Learnability of the superset label learning problem. In *Proceedings of the 31th International Conference on Machine Learning*, pp. 1629–1637, Beijing, China, 2014.

Liu, W., Wang, H., Shen, X., and Tsang, I. W. The emerging trends of multi-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):7955–7974, 2021.

Read, J., Pfahringer, B., Holmes, G., and Frank, E. Classifier chains for multi-label classification. *Machine learning*, 85(3):333–359, 2011.

Saunshi, N., Plevrakis, O., Arora, S., Khodak, M., and Khandeparkar, H. A theoretical analysis of contrastive unsupervised representation learning. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 5628–5637, Long Beach, California, 2019.

Sun, Y., Zhang, Y., and Zhou, Z. Multi-label learning with weak label. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, volume 24, pp. 593–598, Atlanta, Georgia, 2010.

Tishby, N., Pereira, F. C. N., and Bialek, W. The information bottleneck method. *CoRR*, physics/0004057, 2000.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pp. 5998–6008, Long Beach, CA, 2017.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The caltech-ucsd birds-200-2011 dataset. California Institute of Technology, 2011.

Wang, C., Yan, S., Zhang, L., and Zhang, H. Multi-label sparse coding for automatic image annotation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1643–1650, Miami, Florida, 2009.

Wang, K. Robust embedding framework with dynamic hypergraph fusion for multi-label classification. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 982–987, Shanghai, China, 2019.

Wu, B., Liu, Z., Wang, S., Hu, B., and Ji, Q. Multi-label learning with missing labels. In *Proceedings of the 22nd International Conference on Pattern Recognition*, pp. 1964–1968, Stockholm, Sweden, 2014.

Wu, X. and Zhou, Z. A unified view of multi-label performance measures. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 3780–3788, Sydney, NSW, Australia, 2017.

Xie, M.-K., Xiao, J.-H., and Huang, S.-J. Label-aware global consistency for multi-label learning with single positive labels. In *Advances in Neural Information Processing Systems*, volume 35, pp. 18430–18441, New Orleans, LA,, 2022.

Xu, C., Tao, D., and Xu, C. Robust extreme multi-label learning. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1275–1284, San Francisco, CA, 2016.

Xu, M., Jin, R., and Zhou, Z. Speedup matrix completion with side information: Application to multi-label learning. In *Advances in Neural Information Processing Systems 26*, pp. 2301–2309, Lake Tahoe, Nevada, 2013.

Xu, N., Liu, Y.-P., and Geng, X. Label enhancement for label distribution learning. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1632–1643, 2021.

Xu, N., Qiao, C., Lv, J., Geng, X., and Zhang, M.-L. One positive label is sufficient: Single-positive multi-label learning with label enhancement. In *Advances in Neural Information Processing Systems*, pp. 21765–21776, New Orleans, LA, 2022.

Xu, N., Shu, J., Zheng, R., Geng, X., Meng, D., and Zhang, M.-L. Variational label enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (5):6537–6551, 2023.

Yeh, C., Wu, W., Ko, W., and Wang, Y. F. Learning deep latent space for multi-label classification. In *Proceedings of the 21st AAAI Conference on Artificial Intelligence*, pp. 2838–2844, San Francisco, California, 2017.

Yu, H., Jain, P., Kar, P., and Dhillon, I. S. Large-scale multi-label learning with missing labels. In *Proceedings of the 31th International Conference on Machine Learning*, volume 32, pp. 593–601, Beijing, China, 2014.

Yu, Z. and Zhang, M. Multi-label classification with label-specific feature generation: A wrapped approach. *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, 44(9):5199–5210, 2022.

Yun, S., Oh, S. J., Heo, B., Han, D., Choe, J., and Chun, S. Re-labeling imagenet: From single to multi-labels, from global to localized labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2340–2350, virtual, 2021.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *Proceedings of the 5th International Conference on Learning Representations*, Toulon, France, 2017.

Zhang, M.-L. and Zhou, Z.-H. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837, 2013.

Zhou, D., Chen, P., Wang, Q., Chen, G., and Heng, P. Acknowledging the unknown for multi-label learning with single positive labels. In *Proceedings of 17th European Conference on Computer Vision*, volume 13684, pp. 423–440, Tel Aviv, Israel, 2022.

A. Appendix

A.1. Proof of Theorem 4.1

Firstly, the set of hypotheses with expection hamming loss at least ϵ is defined as: $H_\epsilon = \{h \in \mathcal{H} : \mathcal{R}_{\text{ham}}(h) \geq \epsilon\}$. For convenience of proof, we define the dataset $\mathbf{z} = \{\boldsymbol{x}_i, \boldsymbol{y}_i, \boldsymbol{l}_i\}_{i=1}^n$ where $\boldsymbol{y}_i$ and $\boldsymbol{l}_i$ are the ground-truth labels and the pseudo-labels generated by some methods respectively. Let $R_{n,\epsilon}$ be the set of n instances for which there exists an ϵ-bad hypothesis h with zero empirical risk on the dataset with pseudo-labels:

$$R_{n,\epsilon} = \{\mathbf{z} \in (\mathcal{X} \times \mathcal{Y} \times \mathcal{Y})^n : \exists h \in H_\epsilon, \hat{\mathcal{R}}_{\text{ham}}(h) = 0\}. \tag{23}$$

The final objective of this proof is to show that $\Pr(\mathbf{z} \in R_{n,\epsilon}) < \delta$. The proof includes the following two lemmas.

Lemma A.1. *We introduce another dataset $\mathbf{z}'$ of size n, and define the set $S_{n,\epsilon}$ to be the event that there exists a hypothesis in H_ϵ that makes no classification errors on the pseudo-labels of $\mathbf{z}$ but makes at least $\frac{\epsilon}{2}$ classification errors on the ground-truth labels of $\mathbf{z}'$.*

$$S_{n,\epsilon} = \{(\mathbf{z}, \mathbf{z}') \in (\mathcal{X} \times \mathcal{Y} \times \mathcal{Y})^{2n} : \exists h \in H_\epsilon, \hat{\mathcal{R}}_{ham}(h) = 0, \hat{\mathcal{R}}_{\mathbf{z}'}(h) \geq \frac{\epsilon}{2}\},$$

where $\hat{\mathcal{R}}_{\mathbf{z}'}(h) = \frac{1}{nc}\sum_{i=1}^n \sum_{j=1}^c \mathbf{1}(h^j(\boldsymbol{x}_i) \neq y_i^j)$ denote the empirical hamming loss on the ground-truth labels of dataset $\mathbf{z}'$. Then $\Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon}) \geq \frac{1}{2}\Pr(\mathbf{z} \in R_{n,\epsilon})$, when $n > \frac{8}{\epsilon}\ln 2$.

Proof. $\Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon})$ is decomposed as:

$$\Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon}) = \Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon} | \mathbf{z} \in R_{n,\epsilon}) \Pr(\mathbf{z} \in R_{n,\epsilon}).$$

Now we condiser the item $\Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon} | \mathbf{z} \in R_{n,\epsilon})$. Let $H(\mathbf{z}) = \{h \in \mathcal{H} : \hat{\mathcal{R}}_{\text{ham}}(h) = 0\}$ be the set of hypotheses with zero classification errors on the pseudo-labels of $\mathbf{z}$. Then we have:

$$\begin{aligned}
&\Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon} | \mathbf{z} \in R_{n,\epsilon}) \\
&= \Pr\left(\exists h \in H_\epsilon \cup H(\mathbf{z}), \hat{\mathcal{R}}_{\mathbf{z}'}(h) \geq \frac{\epsilon}{2} | \mathbf{z} \in R_{n,\epsilon}\right) \\
&\geq \Pr\left(h \in H_\epsilon \cup H(\mathbf{z}), \hat{\mathcal{R}}_{\mathbf{z}'}(h) \geq \frac{\epsilon}{2} | \mathbf{z} \in R_{n,\epsilon}\right),
\end{aligned}$$

where h is a particular hypothesis in the last line. Since h has error at least ϵ, by adopting Chernoff bound, when $n > \frac{8}{\epsilon}\ln 2$, $\Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon} | \mathbf{z} \in R_{n,\epsilon}) > \frac{1}{2}$. Then the proof is completed. □

With Lemma A.1, we can bound $R_{n,\epsilon}$ by $S_{n,\epsilon}$.

Lemma A.2. *If the hypothesis space $\mathcal{H}$ has Natarajan dimension $d_\mathcal{H}$ and a small unreliability degree $\xi < 1$, then*

$$\Pr((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon}) \leq (2n)^{d_\mathcal{H}} c^{2cd_\mathcal{H}} \exp\left(-\frac{n\theta_1\epsilon}{2}\right).$$

Proof. In the following proof, we consider multi-label learning as a multiple binary classification problem. Then we decompose the dataset $\mathbf{z}$ and $\mathbf{z}'$ as $\mathbf{z} = (\mathbf{x}, \mathbf{y}, \mathbf{l}) = \{\boldsymbol{x}_i, y_i, l_i\}_{i=1}^{nc}$ where y_i is one of the c labels of instance $\boldsymbol{x}_i$ and l_i is the corresponding pseudo-label, each instance is repeated c times in the new dataset because each instance and its corresponding label set is decomposed as c instance-label pairs when we treat it as a multiple binary classification problem. Similarly, $\mathbf{z}' = (\mathbf{x}', \mathbf{y}', \mathbf{l}') = \{\boldsymbol{x}_i', y_i', l_i'\}_{i=1}^{nc}$. For convenience, $j = \pi(y_i) = \pi(l_i)$ is employed to denote the original label index of y_i or l_i.

Next we will do the swapping technique which is used in various proofs of learnability (Liu & Dietterich, 2014). Let the instances from $(\mathbf{x}, \mathbf{y}, \mathbf{l})$ and $(\mathbf{x}', \mathbf{y}', \mathbf{l}')$ form pairs by arbitrary pairing. The two instances of each pair are selected from $(\mathbf{x}, \mathbf{y}, \mathbf{l})$ and $(\mathbf{x}', \mathbf{y}', \mathbf{l}')$ and the two instances are both indexed by a pair index. Define a group G of swaps with

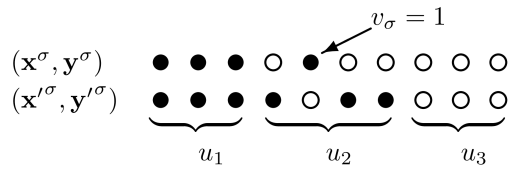


Figure 2: Black dots denote the misclassified instances-labels for a pair of datasets $(\mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}')$ and white dots denote the correctly classified instance-labels.

size $|G| = 2^{nc}$ and a swap $\sigma \in G$ has an index set $J_\sigma \subseteq \{1, \ldots, nc\}$. The result of applying a swap σ is written as $\sigma(\mathbf{z}, \mathbf{z}') = (\mathbf{z}^\sigma, \mathbf{z}'^\sigma)$. Since the swap does not change the measure of exception,

$$\begin{aligned}
&2^{nc} \Pr\left((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon}\right) \\
=&\sum_{\sigma \in G} \mathbb{E}\left[\Pr\left((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon} \mid \mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}'\right)\right] \\
=&\sum_{\sigma \in G} \mathbb{E}\left[\Pr\left(\sigma\left(\mathbf{z}, \mathbf{z}'\right) \in S_{n,\epsilon} \mid \mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}'\right)\right] \\
=&\mathbb{E}\left[\sum_{\sigma \in G} \Pr\left(\sigma\left(\mathbf{z}, \mathbf{z}'\right) \in S_{n,\epsilon} \mid \mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}'\right)\right],
\end{aligned} \tag{24}$$

where the expection comes from the randomness of the generated pseudo-labels $(\mathbf{l}, \mathbf{l}')$ given $(\mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}')$.

Let $\mathcal{H}|(\mathbf{x}, \mathbf{x}')$ be the set of hypothesis making different classifications for $(\mathbf{x}, \mathbf{x}')$. Define set $S^h_{n,\epsilon}$ for each hypothesis $h \in \mathcal{H}$ as:

$$S^h_{n,\epsilon} = \left\{(\mathbf{z}, \mathbf{z}') : \hat{\mathcal{R}}_{\text{ham}}(h) = 0, \hat{\mathcal{R}}_{\mathbf{z}'}(h) \geq \frac{\epsilon}{2}\right\}.$$

By the union bound, we have

$$\begin{aligned}
&\sum_{\sigma \in G} \Pr\left(\sigma\left(\mathbf{z}, \mathbf{z}'\right) \in S_{n,\epsilon} \mid \mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}', \mathbf{y}'\right) \\
\leq &\sum_{h \in \mathcal{H}|(\mathbf{x}, \mathbf{x}')} \sum_{\sigma \in G} \Pr\left(\sigma\left(\mathbf{z}, \mathbf{z}'\right) \in S^h_{n,\epsilon} \mid \mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}'\right).
\end{aligned} \tag{25}$$

Start by expanding the condition in $S^h_{n,\epsilon}$, for a pair of datasets $(\mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}')$. let u_1, u_2 and u_3 represent the number of pairs for which h classifies both incorrectly, one incorrectly, and both correctly. Let $v_\sigma, 0 \leq v_\sigma \leq u_2$ be the number of wrongly-predicted instances swapped into the dataset $(\mathbf{x}^\sigma, \mathbf{y}^\sigma)$, Figure 2 illustrates the situation.

There are $u_1 + u_2 - v_\sigma$ misclassified instances in $(\mathbf{x}', \mathbf{y}')$. $\hat{\mathcal{R}}_{\mathbf{z}'}(h) \geq \frac{\epsilon}{2}$ is equivalent to $u_1 + u_2 - v_\sigma \geq \frac{\epsilon}{2} n$. There are $u_1 + v_\sigma$ misclassified instances in $(\mathbf{x}, \mathbf{y})$. Then:

$$\begin{aligned}
&\Pr\left(\sigma\left(\mathbf{z}, \mathbf{z}'\right) \in S^h_{n,\epsilon} \mid \mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}'\right) \\
=&\mathbf{1}\left(\hat{\mathcal{R}}_{\mathbf{z}'^\sigma}(h) \geq \frac{\epsilon}{2}\right) \prod_{i=1}^{nc} \Pr\left(h^j\left(\boldsymbol{x}_i^\sigma\right) = l_i\right) \\
=&\mathbf{1}\left(\hat{\mathcal{R}}_{\mathbf{z}'^\sigma}(h) \geq \frac{\epsilon}{2}\right) \prod_{i=1}^{nc} \Pr\left(h^j\left(\boldsymbol{x}_i^\sigma\right) = l_i | h^j\left(\boldsymbol{x}_i^\sigma\right) = y_i\right) \mathbf{1}(h^j\left(\boldsymbol{x}_i^\sigma\right) = y_i) \\
&\qquad + \Pr\left(h^j\left(\boldsymbol{x}_i^\sigma\right) = l_i | h^j\left(\boldsymbol{x}_i^\sigma\right) \neq y_i\right) \mathbf{1}(h^j\left(\boldsymbol{x}_i^\sigma\right) \neq y_i) \\
\leq&\mathbf{1}\left(\hat{\mathcal{R}}_{\mathbf{z}'^\sigma}(h) \geq \frac{\epsilon}{2}\right) \prod_{i=1}^{nc} \mathbf{1}(h^j\left(\boldsymbol{x}_i^\sigma\right) = y_i) + \xi \mathbf{1}(h^j\left(\boldsymbol{x}_i^\sigma\right) \neq y_i) \\
\leq&\mathbf{1}\left(u_1 + u_2 \geq \frac{\epsilon}{2} nc\right) \xi^{u_1 + v_\sigma},
\end{aligned} \tag{26}$$

where $j = \pi(l_i)$ and the hypothesis is considered as a binary classification for each label. Then sum up over each swap σ, any swap that switch the instances pairs in $u_1 + u_3$ does not change the bound in Eq. (26). Since $0 \leq v_\sigma \leq u_2$, for each value $0 \leq j \leq u_2$, there are $2^{u_1+u_3} \begin{pmatrix} u_2 \\ j \end{pmatrix}$ swaps that have $v_\sigma = j$, therefore,

$$\begin{aligned}
&\sum_{\sigma \in G} \mathbf{1}\left(u_1 + u_2 \geq \frac{\epsilon}{2}nc\right) \xi^{u_1+v_\sigma} \\
&\leq \mathbf{1}\left(u_1 + u_2 \geq \frac{\epsilon}{2}nc\right) 2^{u_1+u_3} \sum_{j=0}^{u_2} \begin{pmatrix} u_2 \\ j \end{pmatrix} \xi^{u_1+j} \\
&= \mathbf{1}\left(u_1 + u_2 \geq \frac{\epsilon}{2}nc\right) 2^{nc-u_2} \xi^{u_1} (1+\xi)^{u_2} \\
&= \mathbf{1}\left(u_1 + u_2 \geq \frac{\epsilon}{2}nc\right) 2^{nc} \xi^{u_1} \left(\frac{1+\xi}{2}\right)^{u_2}.
\end{aligned} \tag{27}$$

When $(\mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}')$ and h make $u_1 = 0$ and $u_2 = \frac{\epsilon}{2}nc$, the right side reaches its maximum $2^{nc}\left(\frac{1+\xi}{2}\right)^{\frac{n\epsilon c}{2}}$

$$2^{nc} \Pr\left((\mathbf{z}, \mathbf{z}') \in S_{n,\epsilon}\right) \leq (2n)^{d_\mathcal{H}} c^{2cd_\mathcal{H}} 2^{nc} \left(\frac{1+\xi}{2}\right)^{\frac{n\epsilon c}{2}}. \tag{28}$$

Apply the definition of θ_1 to the inequation, completes the proof. □

Proof. Proof of Theorem 4.1
By combining the results of Lemma A.1 and Lemma A.2, we have $\Pr\left(\mathbf{z} \in R_{n,\epsilon}\right) \leq 2^{(d_\mathcal{H}+1)} n^{d_\mathcal{H}} c^{2cd_\mathcal{H}} \exp\left(-\frac{n\theta_1\epsilon}{2}\right)$. Bound this with δ on a log scale to obtain

$$(d_\mathcal{H} + 1)\log 2 + d_\mathcal{H} \log n + 2d_\mathcal{H} c \log c - \frac{n\theta_1\epsilon}{2} \leq \log \delta.$$

Bound $\log n$ with $\left(\log\left(\frac{4d_\mathcal{H}}{\theta_1\epsilon}\right) - 1\right) + \frac{\theta_1\epsilon}{4d_\mathcal{H}} n$, which is usually used as a trick to get a linear form for n. Then we solve for n to obtain the result.

□

A.2. Proof of Theorem 4.2

Proof. Let u_1, u_2 and u_3 represent the number of pairs for which h classifies both incorrectly, one incorrectly, and both correctly. Let $v_\sigma, 1 \leq v_\sigma \leq u_2$ be the number of wrongly-predicted instances swapped into the dataset $(\mathbf{x}^\sigma, \mathbf{y}^\sigma)$.

Let u_1', u_2' and u_3' represent the number of pairs for which h classifies both incorrectly, one incorrectly, and both correctly for the instances with the single positive labels. Let $v_\sigma', 1 \leq v_\sigma' \leq u_2'$ be the number of wrongly-predicted instances swapped into the dataset $(\mathbf{x}^\sigma, \mathbf{y}^\sigma)$.

We consider the extreme case where $\xi = 1$, the pseudo-label can be randomly generated. Similarly to the proof of Theorem 4.1,

$$\begin{aligned}
&\Pr\left(\sigma\left(\mathbf{z}, \mathbf{z}'\right) \in S_{n,\epsilon}^h \mid \mathbf{x}, \mathbf{y}, \mathbf{x}', \mathbf{y}'\right) \\
&= \mathbf{1}\left(R_{\mathbf{z}'^\sigma}(h) \geq \frac{\epsilon}{2}\right) \prod_{i=1}^{nc} \Pr\left(h^j\left(\mathbf{x}_i^\sigma\right) = l_i, j = \pi(l_i)\right) \\
&\leq \mathbf{1}\left(u_1 + u_2 \geq \frac{\epsilon}{2}nc\right) (1-\tau)^{u_1'+v_\sigma'} \xi^{(u_1-u_1')+(v_\sigma-v_\sigma')} \\
&= \mathbf{1}\left(u_1 + u_2 \geq \frac{\epsilon}{2}nc\right) (1-\tau)^{u_1'+v_\sigma'},
\end{aligned} \tag{29}$$

therefore,

$$\begin{aligned}
&\sum_{\sigma\in G}\mathbf{1}\left(u_1+u_2\geq\frac{\epsilon}{2}nc\right)(1-\tau)^{u_1'+v_\sigma'}\\
&\leq\mathbf{1}\left(u_1+u_2\geq\frac{\epsilon}{2}nc\right)2^{u_1+u_3}\sum_{j=0}^{u_2'}\binom{u_2'}{j}(1-\tau)^{u_1'+j}\\
&=\mathbf{1}\left(u_1+u_2\geq\frac{\epsilon}{2}nc\right)2^{nc-u_2}(1-\tau)^{u_1'}(2-\tau)^{u_2'}\\
&=\mathbf{1}\left(u_1+u_2\geq\frac{\epsilon}{2}nc\right)2^{nc}\left(\frac{2-\tau}{2}\right)^{u_2'}\left(\frac{1}{2}\right)^{u_2-u_2'}(1-\tau)^{u_1'}.
\end{aligned}\tag{30}$$

When $(\mathbf{x},\mathbf{y},\mathbf{x}',\mathbf{y}')$ and h make $u_1'=0$ and $u_2=u_2'=\frac{\epsilon}{2}nc$, the right side reaches its maximum $2^{nc}\left(\frac{2-\tau}{2}\right)^{\frac{n\epsilon c}{2}}$. Next proof is same as Theorem 4.1. □

A.3. Detail of Eq. (8)

$$\begin{aligned}
&H(l^j|\boldsymbol{z}^j)\geq H(y^j|\boldsymbol{z}^j)\\
&\Leftrightarrow H(l^j,\boldsymbol{z}^j)-H(\boldsymbol{z}^j)\geq H(y^j,\boldsymbol{z}^j)-H(\boldsymbol{z}^j)\\
&\Leftrightarrow H(l^j,\boldsymbol{z}^j)\geq H(y^j,\boldsymbol{z}^j)\\
&\Leftrightarrow H(\boldsymbol{z}^j)+H(l^j)-I(\boldsymbol{z}^j,l^j)\geq H(\boldsymbol{z}^j)+H(y^j)-I(\boldsymbol{z}^j,y^j)\\
&\Leftrightarrow H(l^j)-I(\boldsymbol{z}^j,l^j)\geq H(y^j)-I(\boldsymbol{z}^j,y^j)\\
&\Leftrightarrow I(\boldsymbol{z}^j,l^j)\leq I(\boldsymbol{z}^j,y^j)\\
&\Leftrightarrow\mathcal{L}_{IB}=\sum_{j=1}^{c}I(\boldsymbol{z}^j,y^j)-\beta_jI(\boldsymbol{z}^j,\boldsymbol{x})\geq\sum_{j=1}^{c}I(\boldsymbol{z}_j^j,l^j)-\beta_jI(\boldsymbol{z}^j,\boldsymbol{x}).
\end{aligned}$$

A.4. Proof of Theorem 5.2

For Eq. (22), let s be the indentified label set; y be a positive label and its index is k then:

$$\begin{aligned}
&\ell(\boldsymbol{x},\boldsymbol{l}_{s\cup\{y\}})-\ell(\boldsymbol{x},\boldsymbol{l}_s)\\
&=\int p(\boldsymbol{z}^k|\boldsymbol{x})\log\frac{p(l^k=1|\boldsymbol{z}^k)}{p(l^k)}d\boldsymbol{z}^k\\
&-\int p(\boldsymbol{z}^k|\boldsymbol{x})\log\frac{p(l^k=0|\boldsymbol{z}^k)}{p(l^k)}d\boldsymbol{z}^k\\
&=\mathbb{E}_{\boldsymbol{z}^k\sim p(\boldsymbol{z}^k|\boldsymbol{x})}\left[\log p(l^k=1|\boldsymbol{z}^k)-\log p(l^k=0|\boldsymbol{z}^k)\right]
\end{aligned}$$

Since the label y is a positive label, $p(l^k=1|\boldsymbol{z}^k)>p(l^k=0|\boldsymbol{z}^k)$. Then $\ell(\boldsymbol{x},\boldsymbol{l}_{s\cup\{y\}})-\ell(\boldsymbol{x},\boldsymbol{l}_s)>0$, there is a gap $\varepsilon>0$ between $\ell(\boldsymbol{x},\boldsymbol{l}_{s\cup\{y\}})$ and $\ell(\boldsymbol{x},\boldsymbol{l}_s)$ for all $0<\varepsilon\leq\min_{1\leq j\leq c}\mathbb{E}_{\boldsymbol{z}^j\sim p(\boldsymbol{z}^j|\boldsymbol{x})}\left[\log\frac{p(l^j=1|\boldsymbol{z}^j)}{p(l^j=0|\boldsymbol{z}^j)}\right]$.

A.5. Implementation Details

In our methods, the encoder of each label-specific feature has the form of $p(\boldsymbol{z}^j|\boldsymbol{x})=\mathcal{N}\left(\boldsymbol{z}^j|f_e^\mu(\boldsymbol{x}),f_e^\Sigma(\boldsymbol{x})\right)$. For the MLIC datasets, f_e consists of a ResNet-50 (He et al., 2016) pretrained on ImageNet and a standard self-attention block (Vaswani et al., 2017) as used in LAGC (Xie et al., 2022). For the MLL datasets, f_e is implemented by a two layer MLP. The decoder $q(\boldsymbol{l}|\boldsymbol{z})$ is a simple linear model followed by a sigmoid function. We use the Adam optimizer (Kingma & Ba, 2015). The batch size is selected from $\{8,16\}$ and the number of epochs is set to 10. The learning rate , weight decay, and the tradeoff parameter β are selected from $\{10^{-2},10^{-3},10^{-4}\}$ with a validation set. All the comparing methods run 5 trials on each datasets. For fairness, we employed ResNet-50 as the backbone for all comparing methods.

Table 5: Characteristics of the MLIC datasets.

Dataset	#Training	#Validation	#Testing	#Classes
VOC	4574	1143	5823	20
COCO	65665	16416	40137	80
NUS	120000	30000	60260	81
CUB	4795	1199	5794	312

Table 6: Characteristics of the MLL datasets.

Dataset	#Examples	#Features	#Classes	#Domain
Image	2000	294	5	Images
Scene	2407	294	6	Images
Yeast	2417	103	14	Biology
Rcv1subset1	6000	944	101	Text
Mediamill	42177	120	101	Video

Table 7: Predictive performance of each comparing methods on MLL datasets in terms of *Hamming loss* (mean ± std). The best performance is highlighted in bold (the smaller the better).

	Image	Scene	Yeast	Rcv1subset1	Mediamill
AN	0.229±0.000	0.161±0.010	0.306±0.000	0.029±0.000	**0.045±0.000**
AN-LS	0.226±0.003	0.160±0.012	0.306±0.000	0.029±0.000	**0.045±0.000**
WAN	0.388±0.069	0.238±0.023	0.267±0.008	0.099±0.001	0.121±0.002
EPR	0.328±0.032	0.222±0.023	0.230±0.003	0.035±0.001	0.048±0.001
ROLE	0.241±0.028	0.150±0.009	0.291±0.005	0.028±0.000	**0.045±0.000**
EM	0.752±0.038	0.722±0.017	0.661±0.045	0.656±0.007	0.715±0.008
EM-APL	0.724±0.094	0.822±0.001	0.680±0.009	0.743±0.010	0.736±0.020
SMILE	0.205±0.008	0.124±0.035	**0.205±0.003**	**0.025±0.000**	0.048±0.002
MIME	**0.188±0.083**	**0.117±0.017**	0.305±0.005	0.028±0.000	**0.045±0.000**

A.6. Details of Datasets

The details of the four MLIC datasets and the five MLL datasets are provided in Table 5 and Table 6 respectively. The basic statics about the MLIC datasets include the number of training set, validation set, and testing set (#Training, #Validation, #Testing), and the number of classes (#Classes). The basic statics about the MLL datasets include the number of examples (#Examples), the dimension of features (#Features), the number of classes (#Classes), and the domain of the dataset (#Domain).

A.7. More Results of MLL Datasets

Table 7 and 8 report the results of our method and other comparing methods on five MLL datasets in terms of *Hamming loss* and *Coverage* respectively.

Table 8: Predictive performance of each comparing methods on MLL datasets in terms of *Coverage* (mean ± std). The best performance is highlighted in bold (the smaller the better).

	Image	Scene	Yeast	Rcv1subset1	Mediamill
AN	0.302±0.043	0.184±0.070	0.486±0.011	0.223±0.002	0.191±0.001
AN-LS	0.247±0.030	0.142±0.025	0.465±0.006	0.158±0.004	0.209±0.005
WAN	0.250±0.027	0.122±0.023	0.460±0.005	0.160±0.002	0.190±0.002
EPR	0.251±0.021	0.150±0.022	0.460±0.002	0.199±0.004	0.197±0.011
ROLE	0.274±0.035	0.134±0.034	0.472±0.007	0.174±0.007	0.238±0.007
EM	0.370±0.025	0.135±0.015	0.529±0.133	0.128±0.001	0.192±0.002
EM-APL	0.375±0.025	0.194±0.040	0.479±0.009	0.144±0.002	0.193±0.003
SMILE	0.177±0.025	0.110±0.032	**0.455±0.071**	**0.121±0.003**	0.198±0.004
MIME	**0.173±0.025**	**0.095±0.009**	0.475±0.023	0.124±0.001	**0.186±0.006**