Complementary to Multiple Labels: A Correlation-Aware Correction Approach

Yi Gao, Miao Xu, and Min-Ling Zhang, *Senior Member, IEEE*

Abstract—*Complementary label learning* (CLL) requires annotators to give *irrelevant* labels instead of relevant labels for instances. Currently, CLL has shown its promising performance on multi-class data by estimating a transition matrix. However, current multi-class CLL techniques cannot work well on multi-labeled data since they assume each instance is associated with one label while each multi-labeled instance is relevant to multiple labels. Here, we show theoretically how the estimated transition matrix in multi-class CLL could be distorted in multi-labeled cases as they ignore co-existing relevant labels. Moreover, theoretical findings reveal that calculating a transition matrix from label correlations in *multi-labeled CLL* (ML-CLL) needs multi-labeled data, while this is unavailable for ML-CLL. To solve this issue, we propose a two-step method to estimate the transition matrix from candidate labels. Specifically, we first estimate an initial transition matrix by decomposing the multi-label problem into a series of binary classification problems, then the initial transition matrix is corrected by label correlations to enforce the addition of relationships among labels. We further show that the proposal is classifier-consistent, and additionally introduce an MSE-based regularizer to alleviate the tendency of BCE loss overfitting to noises. Experimental results have demonstrated the effectiveness of the proposed method.

Index Terms—Complementary label learning, multi-label learning, transition matrix, label correlations.

1 Introduction

In *multi-label learning* (MLL), each instance is associated with a set of *relevant labels*, where the learned classifier aims to predict all relevant labels of unseen instances [1], [2], [3]. MLL is widely used in many real-world applications, such as text categorization [4], [5], image retrieval [6], [7], and medical domain [8], [9], etc. However, collecting precisely multi-labeled data is laborious because of the unknown number of relevant labels per instance and the existence of complex semantic labels [10]. For the example image in Fig. 1, besides the label *Architecture*, there exist other relevant labels whose accurate annotation needs one-by-one checking of the whole label space; in addition, annotators need special geographical and cultural domain knowledge to accurately label the image as *Paris*.

To release the laborious of annotating multi-labeled data, we explore the problem setting of *multi-labeled CLL* (ML-CLL), where each instance is associated with a *single* complementary label (an irrelevant label of the instance) instead of multiple relevant labels. Providing such weakly supervised information will ease the labeling process in large label space because selecting one complementary label is low-cost and requires less domain knowledge than selecting all relevant labels. One example of ML-CLL is given in Fig. 1 when selecting *desert* as the complementary label. Given the complementary label, the goal of ML-CLL is still the same as fully supervised MLL, i.e., learning a model that can accurately predict multiple relevant labels for unseen instances.

Fig. 1. An example of ML-CLL. The relevant labels of the image include *people*, *architecture*, *sky*, *plant*, and *Paris*, indicating that these elements belong to the image. On the other hand, *desert* serves as the complementary label for this image, signifying the absence of a desert in the image. Notably, *Paris* is considered a complex semantic label, as it is difficult to be directly identified without domain knowledge.

The setting of CLL was initially applied in the multi-class learning task [11], [12], [13], [14], [15], [16], [17]. Previous multi-class CLL approaches are based on an estimated transition matrix that summarizes the probability of a label being selected as a complementary label [11], [12], [13]. Although they have achieved a promising performance on multi-class data, they are restricted to the case where an instance is associated with only one relevant label. In this case, multi-class CLL approaches only consider the exclusive relationship among labels, while these approaches ignore that labels can bear other relationships in the multi-labeled case, especially the co-occurrence of labels. In fact, relationships among labels are crucial to solving ML-CLL problems since the selection of a complementary label of an instance in MLL is the combined result against multiple

- *Yi Gao is with the School of Cyber Science and Engineering, Southeast University, Nanjing 210096, China and the Key Laboratory of Computer Network and Information Integration (Southeast University), Ministry of Education, China. E-mail: gao_yi@seu.edu.cn*
- *Miao Xu is with The University of Queensland, Australia. E-mail: miao.xu@uq.edu.au*
- *Min-Ling Zhang (corresponding author) is 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. E-mail: zhangml@seu.edu.cn*

relevant labels rather than against only a single relevant label. Misusing a technique targeting against a single relevant label to the multiple relevant labels case will result in a wrongly estimated transition matrix.

In this paper, we first theoretically analyze how the estimation of the transition matrix using the current multi-class CLL techniques could be distorted in multi-labeled cases. According to these findings, we observe that estimating the transition matrix in ML-CLL from label correlations needs to know relevant labels of instances, while these are unavailable. To remove this pain, we propose a two-step method to estimate the transition matrix in ML-CLL from candidate labels which are the complement of complementary labels. Our strategy includes: (1) estimating an initial transition matrix by decomposing the multi-label problem into binary classification problems; (2) using label correlations to correct the initial transition matrix by enforcing the addition of relationships among labels. The fast convergence of *Cross-Entropy* (CE) loss benefits from focusing on instances that are difficult to classify, which may result in CE loss overfitting to noisy labeled data. As a type of CE loss, *Binary CE* (BCE) loss has the same problem. The study of [18] indicates that *Mean Squared Error* (MSE) loss is less sensitive to noisy labels than CE loss. As *Binary CE* (BCE) loss is a benchmark of our approach, an MSE-based regularizer is further introduced to alleviate the tendency of it overfitting to noises.

In addition, we show that our proposed ML-CLL can be easily combined with learning from relevant labels, which significantly extends the application scenario of the proposed algorithm. This combination is particularly useful, e.g. when labels are collected via crowd-sourcing [19] where crowd-workers are asked to randomly select a complementary label and one or more relevant labels for an instance. Experimental results on various datasets demonstrate the effectiveness of the proposed approach. Especially in situation when each instance is only equipped with a complementary label and a relevant label, our proposal has superior performance, even comparable with the performance on fully supervised data. Our main contributions are summarized as follows:

- We theoretically analyze the distortion of the transition matrix estimated by multi-class CLL in multi-labeled cases, because multi-class CLL techniques ignore the co-existence of relevant labels. Theoretical findings reveal that multi-labeled data is indispensable for calculating the transition matrix from label correlations.
- To solve the problem of unavailable multi-labeled data, we propose a two-step method to estimate the transition matrix from candidate labels. Moreover, we show theoretically that the proposed approach is classifier-consistent under a mild assumption.
- We introduce a practical strategy – MSE-based regularization – to alleviate the overfitting tendency of BCE loss. Our empirical study shows that the proposal obtains comparable performance with state-of-the-art baselines, which proves the effectiveness of our approach.

The rest of this paper are organized as follows. Section 2 briefly reviews related work of ML-CLL. Then we formalize the ML-CLL problem in Section 3, analyze it theoretically and describe our approach in Section 4. In Section 5, we introduce an MSE-based regularization and show how to adapt our method to bear an additional small amount of relevant labels. The experimental results are given in Section 6 and we conclude in Section 7.

2 Related Work

In this section, we will give a brief review of related work of ML-CLL, including MLL, *partial multi-label learning* (PML) and multi-class CLL.

2.1 Multi-Label Learning

MLL problems aim to train a classifier that can predict a set of relevant labels for an unseen instance, where each training instance is associated with multiple relevant labels simultaneously. With the complexity of label correlation, the previous studies can be grouped into three categories [20], [21], [22], [23]: *first-order approach* [24], [25], [26], *second-order approach* [27], [28] and *high-order approach* [29], [30]. To solve MLL problems, the first-order approach decomposes MLL problems into a set of binary classification problems [24], [25]. However, these approaches ignore label correlations among labels, which play a crucial role in MLL [20]. After realizing the importance of label correlation, more and more studies attempt to exploit it to improve MLL performance. Among them, the second-order approach considers the pairwise label correlations that refer to the relationship between two labels. The kind of these approaches generally transform MLL problems into bipartite ranking problems by enforcing that relevant labels should be ranked higher than irrelevant labels [28], [31], [32]. Beyond second-order relationship, there exists more complex relationship between labels in many real-world scenarios. Therefore, many approaches begin to exploit high-order label correlations to handle the MLL problems recently [29], [33], [34], [35]. For example, Zhao et al. [35] leverage variational autoencoder to facilitate the learning process via exploiting high-order correlations among labels, while Wang et al. and Xun et al. [36], [37] both design special neural network blocks to automatically extract label correlations to improve the label prediction performance. Although high-order approaches have the ability of stronger label correlation-modeling, they may suffer from high computational cost comparing to first and second-orders approaches [38].

2.2 Partial Multi-Label Learning

Due to that the fully supervised data is difficult to collect, many researchers tend to explore the weakly supervision data form to alleviate the heavy load of labeled data collection [39]. PML is a recently emerging weakly supervised approch firstly proposed by Xie et al. [40]. In PML, each training instance is associated with a set of candidate labels that consist of *relevant* labels and *irrelevant* (noisy) labels and the goal is to learn a classifier assigning a set of labels accurately for unseen instances.

At the first glance, it seems that ML-CLL is an extreme case of PML, such that all PML methods are also applicable to ML-CLL. However, existing PML methods assume that

TABLE 1
Summary of major mathematical notations.

Notations	Mathematical meanings
$\mathcal{X}$	the feature space of instances
$\mathcal{Y}$	the label space with K possible labels $\{l_1, l_2, \ldots, l_K\}$
$\boldsymbol{x}$	a multi-label instance ($\boldsymbol{x} \in \mathcal{X}$)
Y	the relevant label set of $\boldsymbol{x}$ ($Y \subset \mathcal{Y}, Y \neq \emptyset$ nor $\mathcal{Y}$)
$\boldsymbol{y}$	the binary vector of Y ($\boldsymbol{y} = [y^1, y^2, \ldots, y^K]$), where $y^k = 1$ indicates that $l_k \in Y$ and 0 otherwise
D	multi-label dataset $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, Y)$
$\boldsymbol{f}(\cdot)$	real-valued decision function $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^K$, where $f^k(\boldsymbol{x})$ predicts label l_k being the relevant label of $\boldsymbol{x}$
$L(\cdot, \cdot)$	proper MLL loss function
$R_L(\boldsymbol{f})$	the expected risk of MLL
$\boldsymbol{f}^*$	the minimizer of $R_L(\boldsymbol{f})$ ($\boldsymbol{f}^* = \operatorname{argmin}_{\boldsymbol{f}} R_L(\boldsymbol{f})$)
$\bar{y}$	the complementary label of $\boldsymbol{x}$ ($\bar{y} \in \mathcal{Y} \setminus Y$)
$\bar{\boldsymbol{y}}$	K-dimensional vector of $\bar{y}$ ($\bar{\boldsymbol{y}} = [\bar{y}^1, \bar{y}^2, \ldots, \bar{y}^K]$), where $\bar{y}^j = 1$ indicates $\bar{y} = l_j$ and 0 otherwise
$\widehat{Y}$	the candidate label set of $\boldsymbol{x}$, where $\widehat{Y} = \mathcal{Y} \setminus \bar{y}$
$\widehat{\boldsymbol{y}}$	K-dimensional vector of $\widehat{Y}$, where $\widehat{\boldsymbol{y}} = \mathbf{1} - \bar{\boldsymbol{y}}$
$\bar{D}$	ML-CLL training set $\{(\boldsymbol{x}_i, \bar{y}_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, \bar{y})$
$\bar{L}(\cdot, \cdot)$	ML-CLL loss function
$R_{\bar{L}}(\boldsymbol{f})$	the expected risk of ML-CLL
$\mathcal{Y}'$	the label space excludes $\emptyset$ and $\mathcal{Y}$, where $\mathcal{Y}' = \{2^{\mathcal{Y}} - \emptyset - \mathcal{Y}\}$
C	a subset of $\mathcal{Y}'$ ($C \in \mathcal{Y}'$), where C_k is the k-th label subset in $\mathcal{Y}'$
$\tilde{\mathbf{T}}$	high-dimension transition matrix, $\tilde{\mathbf{T}} \in \mathbb{R}^{(2^K-2)\times K}$
$\mathbf{T}$	low-dimension transition matrix $\mathbf{T} \in [0,1]^{K\times K}$ used to replace $\tilde{\mathbf{T}}$
$\mathbf{Q}$	the transition matrix estimated in multi-class CLL, where $\mathbf{Q} \in [0,1]^{K\times K}$
ℓ_j	the difference between $\mathbf{T}$ and $\mathbf{Q}$ on label l_j being the complementary label of $\boldsymbol{x}$
$\mathbf{S}$	initial transition matrix, where $\mathbf{S} \in [0,1]^{K\times K}$
A_k	the subset of $\boldsymbol{x}$ in $\bar{D}$ with $\bar{y}^k = 1$
$\mathbf{C}$	the label correlation matrix ($\mathbf{C} \in [0,1]^{K\times K}$), which presents the relationship among labels
$\widehat{\mathbf{T}}$	the estimation of $\mathbf{T}$, where $\widehat{\mathbf{T}}$ is calculated by the proposed method in the paper
$\boldsymbol{f}^*_{CL}$	the minimizer of $R_{\bar{L}}(\boldsymbol{f})$ ($\boldsymbol{f}^*_{CL} = \operatorname{argmin}_{\boldsymbol{f}} R_{\bar{L}}(\boldsymbol{f})$)
$\bar{L}_{mse}(\cdot, \cdot)$	the MSE-based regularizer
$\bar{\mathcal{L}}(\cdot, \cdot)$	the target loss function in this paper
$\tilde{Y}$	the subset of Y, where $\tilde{Y} \subseteq Y$ and $\tilde{Y} \neq \emptyset$
$\tilde{\boldsymbol{y}}$	binary vector of $\tilde{Y}$, where $\tilde{y}^1 = 1$ when label $l_k \in \tilde{Y}$ and 0 otherwise

noisy labels only compose a small portion in the candidate labels [38], [41], [42], [43], such that many approaches [38], [42], [43] adopt matrix factorization to tackle PML problems, which decompose the candidate label matrix into the low-rank multi-label matrix and the sparse noisy label matrix. Compared to PML, the studied ML-CLL problem in this paper are targeted at the problem with only one complementary label, resulting in a high-noise PML problem on which the existing approaches can not be applicable. We will demonstrate the performance difference in the experimental part.

2.3 Multi-Class Complementary Label Learning

Currently, CLL problem is only considered in multi-class learning, whose goal is to predict a single relevant label per instance precisely from complementary labeled data. Previous approaches can be roughly grouped into two categories: (1) modeling the generative relationship between the complementary label and the relevant label [11], [12], [13], [17], [44]; (2) modeling the probability of complementary labels from the learned discriminative classifier directly [14], [15], [16].

The first multi-class CLL method belongs to category one. It models the generative relationship between complementary labels and relevant labels, and uses a such generative process to rewrite one-versus-all and pairwise comparison loss functions to derive an unbiased risk estimator [11]. Ishida et al. [12] realize that the method of [11] is restricted to loss functions and propose a new method which can use arbitrary losses and models. A typical way to make use of the modeled generative process is through a transition matrix, which summarizes the probabilities of a label being complementary labels when relevant labels are given. Then, approaches apply a transition matrix to recover relevant labels from complementary labels [12], [13], [44]. Compared with [11], [12], transition matrix-based methods can map more complex generative relationship rather than uniform one only. Therefore, we tend to design a transition matrix-based method to solve ML-CLL problem with a different estimating way.

Differ from category one, approaches residing in category two directly model the probabilities of complementary labels from the learned classifier without the generative relationship [14], [15], [16]. Chou et al. propose a surrogate complementary loss framework based on complementary labels providing negative feedback during the training process [14]. Although its losses fail to derive an unbiased risk estimator, it achieves good performance on the multi-class CLL. In light of the property of the complementary label that the predictive probability of the complementary label is expected to approach zero, [15] and [16] propose a discriminative solution by directly modeling the probabilities of complementary labels from learned classifier to avoid the generative assumption. Due to that multi-class CLL approaches are designed for a single relevant label case, which are not suitable for the ML-CLL case that an instance is associated with multiple labels simultaneously. We will demonstrate that in the experimental part.

3 Problem Setup

In MLL, let $\mathcal{X}$ be the feature space and $\mathcal{Y} = \{l_1, l_2, \ldots, l_K\}$ be the finite label space with K possible class labels ($K > 2$). A multi-label instance $\boldsymbol{x} \in \mathcal{X}$ is equipped with a set of relevant labels $Y \subset \mathcal{Y}$. $(\boldsymbol{x}, Y)$ is independently sampled from an unknown joint probability distribution $p(\boldsymbol{x}, Y)$. Here we exclude the special cases of $Y = \emptyset$ nor $\mathcal{Y}$ to ensure relevant labels and complementary labels both exist. For convenience, we use a binary vector $\boldsymbol{y} = [y^1, y^2, \ldots, y^K] \in \{0,1\}^K$ to denote Y, where $y^k = 1$ indicates that $l_k \in Y$ is relevant to $\boldsymbol{x}$ and 0 otherwise. Suppose $D = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, Y)$ is the training set with n instances. The goal of MLL is to learn a multi-label classifier $h : \mathcal{X} \to 2^{\mathcal{Y}}$, which can predict a set of relevant labels for any unseen instance. Instead of learning h directly, most MLL methods tend to learn a real-

valued decision function $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^K$ via minimizing the expected risk

$$R_L(\boldsymbol{f}) = \mathbb{E}_{p(\boldsymbol{x},Y)}[L(\boldsymbol{f}(\boldsymbol{x}), \boldsymbol{y})], \tag{1}$$

where L is a proper MLL loss function [35], such as BCE loss. $\boldsymbol{f}(\boldsymbol{x})$ is usually interpreted as a probability vector: $f^k(\boldsymbol{x})$ is the k-th entry of $\boldsymbol{f}(\boldsymbol{x})$ and predicts the confidence score that label l_k is relevant to $\boldsymbol{x}$, i.e., if properly normalized then $p(y^k = 1|\boldsymbol{x})$. Denoting the optimal classifier learned from the expected risk as $\boldsymbol{f}^*$, i.e., $\boldsymbol{f}^* = \operatorname{argmin}_{\boldsymbol{f}} R_L(\boldsymbol{f})$.

In ML-CLL studied in this paper, each training instance is equipped with a *single* complementary label. The complementary labeled instance $(\boldsymbol{x}, \bar{y}) \in (\mathcal{X}, \mathcal{Y})$ is drawn from an unknown joint probability distribution $p(\boldsymbol{x}, \bar{y})$, where $\bar{y} \in \mathcal{Y} \setminus Y$ is a complementary label of $\boldsymbol{x}$. $\bar{y}$ can be presented as a K-dimensional vector $\bar{\boldsymbol{y}} = [\bar{y}^1, \bar{y}^2, \ldots, \bar{y}^K]$. If label l_j is selected as the complementary label to $\boldsymbol{x}$ ($\bar{y} = l_j$), then $\bar{y}^j$ is one and all other elements are zero in $\bar{\boldsymbol{y}}$. We utilize $\widehat{Y} = \mathcal{Y} \setminus \bar{y}$ to denote *the candidate label set* of $\boldsymbol{x}$. Let $\widehat{\boldsymbol{y}} = [\widehat{y}^1, \widehat{y}^2, \ldots, \widehat{y}^K]$ to be the corresponding vector representation of subset $\widehat{Y}$, where all elements are one except for the one corresponding to the complementary label, which is set to zero ($\widehat{\boldsymbol{y}} = \mathbf{1} - \bar{\boldsymbol{y}}$).

Let $\bar{D} = \{(\boldsymbol{x}_i, \bar{y}_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} p(\boldsymbol{x}, \bar{y})$ be the ML-CLL training set with n instances. The expected risk of multi-labeled CLL is defined over $p(\boldsymbol{x}, \bar{y})$:

$$R_{\bar{L}}(\boldsymbol{f}) = \mathbb{E}_{p(\boldsymbol{x},\bar{y})}[\bar{L}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}})], \tag{2}$$

where $\bar{L}$ denotes a ML-CLL loss, which will be proposed later this paper. For convenient reference, Table 1 provides a summary of the main notations used in this paper, along with their corresponding mathematical interpretations.

4 The Proposed Approach

In this section, we first introduce the definition of the transition matrix in MLL and analyze why the estimated transition matrix using multi-class techniques is unsuitable for ML-CLL. Then, we describe an advanced two-step way to estimate the transition matrix in the MLL case. Finally, we prove our approach is classifier-consistent with a mild assumption.

4.1 Transition Matrix for ML-CLL

In ML-CLL, we start by introducing a transition matrix $\tilde{\mathbf{T}}$ that summarizes the probabilities for a complementary label given a set of relevant labels. More specifically, the transition matrix $\tilde{\mathbf{T}}$ is defined as $\tilde{\mathbf{T}}_{kj} = p(\bar{y}^j = 1|Y = C_k)$ where $C_k \in \mathcal{Y}' = \{2^{\mathcal{Y}} - \emptyset - \mathcal{Y}\}$ ($k \in [2^K - 2]$) is the k-th label subset. If $l_j \in C_k$, then $\tilde{\mathbf{T}}_{kj} = 0$ because the label l_j has no chance to be selected as the complementary label. In this paper, we employ the same class-dependent assumption as the multi-class CLL approach [13]: $p(\bar{y}|Y, \boldsymbol{x}) = p(\bar{y}|Y)$ as $\bar{y}$ and $\boldsymbol{x}$ are conditionally independent given Y. Then we can obtain the following equation:

$$p(\bar{y}^j = 1|\boldsymbol{x}) = \sum_{C \in \mathcal{Y}', l_j \notin C} p(\bar{y}^j = 1|Y = C) p(Y = C|\boldsymbol{x}), \tag{3}$$

where we assume the label l_j is a complementary label of $\boldsymbol{x}$. Then, according to Eq. (3), $p(\bar{y}|\boldsymbol{x})$ can be approximated by $p(Y|\boldsymbol{x})$ when the transition matrix $\tilde{\mathbf{T}}$ is known. If considering all possible label subsets of $\mathcal{Y}'$ as C, we have $\tilde{\mathbf{T}} \in \mathbb{R}^{(2^K-2)\times K}$, i.e., the size of $\tilde{\mathbf{T}}$ depends on the size of the power set of $\mathcal{Y}'$. computing and storing the power set of $\mathcal{Y}'$ would be infeasible due to its exponential growth, especially when K is large. For example, with a large number of possible labels, $2^K - 2$ becomes excessively large. To solve this combinatorial explosion problem, we explore a more practical way to use an alternative lower-dimensional transition matrix to replace the higher-dimensional one. We start investigating the feasibility of the alternative lower-dimensional matrix from Theorem 1.

Theorem 1. *Given an instance $\boldsymbol{x}$, suppose Y is the relevant label set and the label l_j is the complementary label which is randomly selected. Then the following equality holds:*

$$\begin{aligned} p(\bar{y}^j = 1|\boldsymbol{x}) &= \sum_{C \in \mathcal{Y}', l_j \notin C} p(\bar{y}^j = 1|Y = C) p(Y = C|\boldsymbol{x}) \\ &\geq \sum_{k=1, k\neq j}^{K} p(\bar{y}^j = 1|y^k = 1) p(y^k = 1|\boldsymbol{x}). \end{aligned}$$

The second inequality holds because of addition rule of probability. The detailed proof is in Appendix A. Theorem 1 shows that using $\mathbf{T}$ to approximate $p(\bar{\boldsymbol{y}}|\boldsymbol{x})$ is a lower bound of using $\tilde{\mathbf{T}}$ to approximate $p(\bar{\boldsymbol{y}}|\boldsymbol{x})$. Observed by Eq. (3), we find that our main goal transforms from precisely predicting the relevant label set Y of $\boldsymbol{x}$ to precisely predicting its complementary label $\bar{y}$ via the transition matrix $\tilde{\mathbf{T}}$. This means that we need to maximize the predictive probability of the complementary label of $\boldsymbol{x}$, i.e., maximizing $p(\bar{y}|\boldsymbol{x})$. From this point of view, Theorem 1 theoretically shows the feasibility of using a low-dimension transition matrix to replace the high-dimension $\tilde{\mathbf{T}}$, because we optimize by maximizing the lower bound of Eq. (3). Let $\mathbf{T} \in [0, 1]^{K\times K}$ denote the lower-dimensional transition matrix, where the (k, j)-th element of $\mathbf{T}$ is $\mathbf{T}_{kj} = p(\bar{y}^j = 1|y^k = 1)$, and $\mathbf{T}_{kj} = 0$ when $k = j$. Thus, we adopt the $K \times K$ matrix $\mathbf{T}$ as the transition matrix in the following of the paper to avoid the pain in computation and storage brought up by the $(2^K - 2) \times K$ matrix $\tilde{\mathbf{T}}$.

4.2 Distortion in Estimating the Transition Matrix

Before exploring how the transition matrix estimated by multi-class CLL is distorted from that of ML-CLL, we first introduce the transition matrix estimated by multi-class CLL techniques. Suppose $\mathbf{Q} \in [0, 1]^{K\times K}$ be the transition matrix estimated in multi-class CLL. Recalling the approach [13], it estimates the transition matrix under a special assumption: for each label l_k, existing an anchor set $\mathcal{S}_{\boldsymbol{x}|l_k} \subset \mathcal{X}$ such that $p(y^k = 1|\boldsymbol{x}) = 1$ and $p(y^{k'} = 1|\boldsymbol{x}) = 0$ ($l_{k'} \in \mathcal{Y} \setminus \{l_k\}$). With this assumption and regardless of label correlations, the estimation of $\mathbf{Q}_{kj}$ is $p(\bar{y}^j = 1|y^k = 1) = p(\bar{y}^j = 1|\boldsymbol{x})$ iff $\boldsymbol{x}$ is sampled from $\mathcal{S}_{\boldsymbol{x}|l_k}$, where $\mathbf{Q}_{kj}$ is the k-th row and j-th column element of $\mathbf{Q}$.

To measure the distortion between $\mathbf{T}$ calculated in ML-CLL and the estimated $\mathbf{Q}$, we define their difference on the complementary label l_j of $\boldsymbol{x}$ as follows

$$\ell_j = \sum_{k=1}^{K} |\mathbf{T}_{kj} - \mathbf{Q}_{kj}|. \tag{4}$$

The larger value of $\sum_{j=1}^{K} \ell_j$ indicates that $\mathbf{T}$ deviates further from $\mathbf{Q}$. As we know, label correlations and co-occurred multiple labels are key properties of MLL. Due to that the correlations among labels are intricate, directly calculating $\mathbf{T}$ from all label correlations will bring high computational cost. For convenience, we give a simple case of MLL including label correlations – at most two labels can co-occur for an instance, and the rest of labels are mutually exclusive – to facilitate us calculating $\mathbf{T}$ from label correlations and explore the distortion of $\mathbf{T}$ and $\mathbf{Q}$. We start to study the above contents from the definition of mutually exclusive.

Definition 2. *A label set $\mathcal{Y}$ is* **mutually exclusive** *if, for every $\boldsymbol{x} \in \mathcal{X}$, only one element of $\mathcal{Y}$ is relevant to $\boldsymbol{x}$. In other words, if $y_1 \in \mathcal{Y}$ is the relevant label for $\boldsymbol{x}$, then no other label in $\mathcal{Y} \setminus \{y_1\}$ is relevant to $\boldsymbol{x}$.*

Under the simple case in MLL, in Theorem 3, we state how to estimate $\mathbf{T}$ directly from label correlations, and the distortion of $\mathbf{T}$ and $\mathbf{Q}$.

Theorem 3. *Under a MLL scenario: suppose the labels $l_{z_1}, l_{z_2} \in \mathcal{Y}$ ($z_1, z_2 \in [K], z_1 \neq z_2$) are dependent, and the labels belonging to $\mathcal{Y} \setminus \{l_{z_1}, l_{z_2}\}$ are mutually exclusive. For any $\boldsymbol{x}$, its label set $Y \subseteq \{l_{z_1}, l_{z_2}\}$ and $Y \neq \emptyset$. Let the label l_j ($j \in [K], j \neq z_1, z_2$) be the complementary label of $\boldsymbol{x} \in \mathcal{X}$. $\mathbf{T}_{z_1 j}$ and $\mathbf{T}_{z_2 j}$ calculated from label correlations satisfy*

$$\mathbf{T}_{z_1 j} = \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x}) p(y^{z_1} = 1|\boldsymbol{x})},$$
$$\mathbf{T}_{z_2 j} = \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x}) p(y^{z_2} = 1|\boldsymbol{x})},$$

where $[K]$ denotes the integer set $\{1, 2, \ldots, K\}$. The difference of $\mathbf{T}$ and $\mathbf{Q}$ on the complementary label l_j is

$$\ell_j \geq 2(\frac{1}{\xi^2} - 1) p(\bar{y}^j = 1|\boldsymbol{x}),$$

where $\xi = \max\{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x}), p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x})\}$.

The proof is provided in Appendix B. From Theorem 3, we can see that calculating the transition matrix from label correlations is more complex than estimating one without label correlations, and the relevant label sets of instances need to be known. Moreover, Theorem 3 shows that there is a distortion between $\mathbf{T}$ and $\mathbf{Q}$, which widely exists in multi-labeled cases since each multi-label instance is relevant to multiple labels. The above learning scenario only considers the pairwise label correlations, while there exists a more complex relationship among labels. Similarly, under a realizable computational cost, we construct another simple MLL scenario with more complex label relationships to explore factors that affect ℓ_j in Corollary 4.

Corollary 4. *Under a MLL scenario: there are m ($m \geq 2$) labels $l_{z_1}, l_{z_2}, \ldots, l_{z_m} \in \mathcal{Y}$ ($z_1, \ldots, z_m \in [K]$) that are dependent, while the labels belong to $\mathcal{Y} \setminus \{l_{z_1}, l_{z_2}, \ldots, l_{z_m}\}$ are mutually exclusive. For any $\boldsymbol{x} \in \mathcal{X}$, its relevant set $Y \subseteq \{l_{z_1}, l_{z_2} \ldots, l_{z_m}\}$ and $Y \neq \emptyset$. Suppose the label l_j is the complementary label of $\boldsymbol{x}$. The difference ℓ_j between $\mathbf{T}$ and $\mathbf{Q}$ has*

$$\ell_j \geq m(\frac{1}{\xi^m} - 1) p(\bar{y}^j = 1|\boldsymbol{x}),$$

where $\xi = \max\{p(y^{z_m} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{m-1}} = 1, \boldsymbol{x}), p(y^{z_{m-1}} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{m-2}} = 1, y^{z_m} = 1, \boldsymbol{x}), \ldots, p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \ldots, y^{z_m} = 1, \boldsymbol{x})\}$ ($\xi \in (0, 1]$).

The proof is shown in Appendix C. According to Corollary 4, when label correlations are more complex, the distortion of the transition matrix estimated by the multi-class CLL approach is more serious as m increases. Meanwhile, it demonstrates that the ML-CLL problem cannot be solved by current techniques in multi-class CLL.

4.3 Estimation $\mathbf{T}$ with Label Correlations

As discussed above, calculating the transition matrix $\mathbf{T}$ from label correlations needs instances whose relevant label sets are known. Moreover, calculating $\mathbf{T}$ is more and more difficult as relationships among labels become more complex by observing the results of $\mathbf{T}$ in Theorem 3 and Corollary 4. Due to that multi-labeled data are unavailable for our setting, we propose a two-step method to estimate $\mathbf{T}$ from candidate labels, and it can reduce the complexities in calculating $\mathbf{T}$ from label correlations. This two-step method includes: (1) computing an initial transition matrix $\mathbf{S} \in [0, 1]^{K \times K}$ from candidate labels by decomposing the multi-label problem into a series of binary classification problem; (2) obtaining the final estimation of $\mathbf{T}$ by using label correlations to correct $\mathbf{S}$.

Computing an initial transition matrix S. Let $\mathbf{S}_{kj} = p(\bar{y}^j = 1|\widehat{y}^k = 1)$ be an initial transition probability, which is a (k, j)-th element of $\mathbf{S}$. We caulculate $\mathbf{S}$ from candidate labels of instances. Multiplication theorem of probability [1] is applied to calculate $\mathbf{S}_{kj}$ and ensure that the following equation holds:

$$\begin{aligned} \mathbf{S}_{kj} &= p(\bar{y}^j = 1|\widehat{y}^k = 1) \\ &= p(\bar{y}^j = 1|\widehat{y}^k = 1) \int p(\boldsymbol{x}|\bar{y}^j = 1, \widehat{y}^k = 1) d\boldsymbol{x} \\ &= \int p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x}) p(\boldsymbol{x}|\widehat{y}^k = 1) d\boldsymbol{x} \\ &= \mathbb{E}_{p(\boldsymbol{x}|\widehat{y}^k=1)}[p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x})], \end{aligned} \tag{5}$$

where $j, k \in [K]$ and $j \neq k$. In practice, $\mathbb{E}_{p(\boldsymbol{x}|\widehat{y}^k=1)}[p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x})]$ can be approximated by the expectation of $p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x})$ over the conditional distribution $p(\boldsymbol{x}|\widehat{y}^k = 1)$. Assuming $\bar{y}$ and $\widehat{Y}$ are conditionally independent given $\boldsymbol{x}$, so $p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x}) = p(\bar{y}^j = 1|\boldsymbol{x})$. Intuitively, $p(\bar{y}^j = 1|\boldsymbol{x})$ can be approximated by the classifier learned from $\bar{D}$ to predict the probability of complementary labels. Let A_k denote the subset of $\boldsymbol{x}$ in $\bar{D}$ with $\widehat{y}^k = 1$,

1. $p(\boldsymbol{x}, \bar{y}^j = 1, \widehat{y}^k = 1) = p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x}) p(\boldsymbol{x}|\widehat{y}^k = 1) p(\widehat{y}^k = 1) = p(\bar{y}^j = 1|\widehat{y}^k = 1) p(\boldsymbol{x}|\bar{y}^j = 1, \widehat{y}^k = 1) p(\widehat{y}^k = 1) \Rightarrow p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x}) p(\boldsymbol{x}|\widehat{y}^k = 1) = p(\bar{y}^j = 1|\widehat{y}^k = 1) p(\boldsymbol{x}|\bar{y}^j = 1, \widehat{y}^k = 1)$

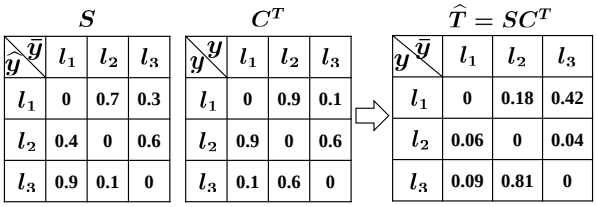

Fig. 2. An example of correcting S with label correlations.

which satisfies the conditional distribution $p(\boldsymbol{x}|\widehat{y}^k = 1)$. Thus, $\mathbf{S}_{kj}$ can be estimated by

$$\begin{aligned}\mathbf{S}_{kj} &= \frac{1}{|A_k|}\sum_{\boldsymbol{x}\in A_k} p(\bar{y}^j = 1|\widehat{y}^k = 1, \boldsymbol{x}) \\ &= \frac{1}{|A_k|}\sum_{\boldsymbol{x}\in A_k} p(\bar{y}^j = 1|\boldsymbol{x}).\end{aligned} \tag{6}$$

Estimating $\mathbf{T}$ with label correlations. The calculating procedure of $\mathbf{S}$ lacks exactly supervised data. Observed by the transition probabilities of $\mathbf{T}$ calculated from label correlations in subsection 4.2, we can find that they are affected by label correlations. Moreover, a label that is low-co-occurred to the relevant labels could be preferentially selected as the complementary label from the view of label correlations. For example, considering *water* as the relevant label; in this case, *desert* (low-co-occurred label) will have a larger chance to be selected as the complementary label compared to *fish* (high-co-occurred label). Motivated by these findings, we use label correlations to correct the initial matrix $\mathbf{S}$ to estimate $\mathbf{T}$ by enforcing the addition of relationships among labels.

Suppose $\mathbf{C} \in [0,1]^{K\times K}$ be a label correlation matrix, where the element $\mathbf{C}_{kj}$ represents the correlation between labels l_k and l_j. The value of $\mathbf{C}_{kj}$ is larger when the correlation of labels l_k and l_j is stronger. Following [40], [45], we adopt the co-occurrence rate of two candidate labels as their correlations. Finally, the transition matrix $\mathbf{T}$ can be estimated by $\widehat{\mathbf{T}} = \mathbf{S}\mathbf{C}^T$, where $\widehat{\mathbf{T}}_{kj} = 0$ if $k = j$, and normalizing $\mathbf{T}$ by row.

Fig. 2 is an example of refining procedure. As can be seen from the Fig. 2, though the estimated initial probability of $p(\bar{y}^2 = 1|\widehat{y}^1 = 1)$ is higher than $p(\bar{y}^3 = 1|\widehat{y}^1 = 1)$ in $\mathbf{S}$, the value of $p(\bar{y}^2 = 1|y^1 = 1)$ is lower than $p(\bar{y}^3 = 1|y^1 = 1)$ in $\widehat{\mathbf{T}}$. This is because the labels l_1 and l_2 have a strong correlation as shown in $\mathbf{C}$, so the label l_2 has a lower chance to be selected as the complementary label for the label l_1. The corrected initial transition matrix $\mathbf{S}$ agrees with our expectation on the low-co-occurred labels that tend to be selected as complementary labels preferentially. In practice, the estimation of $\mathbf{T}$ depends on $p(\bar{y}|\boldsymbol{x})$, where the classifier should perfectly model the probability of complementary labels. When data equipped with complementary labels is sufficiently, the perfect model is capable of modeling $p(\bar{y}|\boldsymbol{x})$.

4.4 A Classifier-Consistent Approach

According to the transition matrix $\mathbf{T}$, we can derive the probability of complementary labels from multi-label classifier. Let $\bar{\boldsymbol{f}}(\boldsymbol{x}) \in \mathbb{R}^K$ be a complementary label classifier, which is defined as

$$\bar{\boldsymbol{f}}(\boldsymbol{x}) = \mathbf{T}^T\boldsymbol{f}(\boldsymbol{x}), \tag{7}$$

Algorithm 1: MLCL Algorithm

Input:
$\bar{D}$: the complementary-label training set $\{(\boldsymbol{x}_i, \bar{\boldsymbol{y}}_i)\}_{i=1}^n$;
E: the number of epochs;
$\mathcal{A}$: an external stochastic optimization algorithm;
Output:
θ: model parameter for $\boldsymbol{f}(\boldsymbol{x};\theta)$;

```
if T is unknown then
    Train a classifier f̄(x) with the softmax output layer and Cross-Entropy loss on D̄;
    Fill S ∈ [0,1]^{K×K} with zeros;
    for k = 1 to K do
        num = 0;
        for (x_i, ȳ_i) ∈ D̄ such that ȳ_i^k = 0 do
            num += 1;
            S_{k·} += f̄(x_i); //add f̄(x_i) to k-th row of S
        end
        S_{k·} /= num;
    end
    T̂ = SC^T;
end
for t = 1 to E do
    Let L be the risk, L = (1/n) Σ_{i=1}^n L̄(f(x_i), ȳ_i) =
      (1/n) Σ_{i=1}^n (L(T̂^T f(x_i), ȳ_i) + ||ȳ_i − T̂^T f(x_i)||_F^2);
    Set gradient −∇_θ L;
    Update θ by A;
end
```

where $\bar{\boldsymbol{f}}(\boldsymbol{x})$ is applied to approximate $p(\bar{\boldsymbol{y}}|\boldsymbol{x})$, $\bar{f}^j(\boldsymbol{x})$ refers to the j-th element of $\bar{\boldsymbol{f}}(\boldsymbol{x})$. ML-CLL problems aim to recover a set of relevant labels per instance from a complementary label. Since training instances are associated with complementary labels, the common loss functions of MLL are unsuitable for ML-CLL. Therefore, we define a complementary loss function $\bar{L}$ as

$$\bar{L}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) = L(\bar{\boldsymbol{f}}(\boldsymbol{x}), \bar{\boldsymbol{y}}) = L(\mathbf{T}^T\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}). \tag{8}$$

Denote by $\boldsymbol{f}^*_{\bar{C}L}$ the minimizer of $R_{\bar{L}}(\boldsymbol{f})$. Recalling the definition of classifier-consistent, if a classifier learned by an approach finally converges to the optimal classifier $\boldsymbol{f}^*$ learned in MLL as the number of instances increases, then this approach is classifier-consistent [46], [47], [48]. We derive that our proposal is classifier-consistent based on a mild assumption:

Assumption 5. *Suppose the transition matrix $\mathbf{T}$ is invertible and can perfectly recover the relationship between relevant labels of $\boldsymbol{x}$ and its complementary label. Then, we have $\bar{\boldsymbol{y}} = \mathbf{T}^T\boldsymbol{y}$.*

With Assumption 5, our approach trained on $\bar{L}$ can be inferred to be classifier-consistent, which is stated in Theorem 6. Naturally, Theorem 6 guarantees that the optimal classifier learned from complementary labeled data converges to the optimal one learned from fully supervised MLL.

Theorem 6. *With Assumption 5, suppose the transition matrix $\mathbf{T}$ is invertible, then the ML-CLL optimal classifier $\boldsymbol{f}^*_{\bar{C}L}$ converges to the MLL optimal classifier $\boldsymbol{f}^*$.*

The proof is represented in Appendix D. Thanks to BCE loss is a popular loss function in MLL, we adopt BCE loss as the base in this paper, then $\bar{L}$ is expressed as

$$\bar{L}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) = -\bar{\boldsymbol{y}}\log(\mathbf{T}^T\boldsymbol{f}(\boldsymbol{x})) - (\mathbf{1}-\bar{\boldsymbol{y}})\log(\mathbf{1}-\mathbf{T}^T\boldsymbol{f}(\boldsymbol{x}))), \tag{9}$$

where $\mathbf{1}$ denotes a K-dimensional vector with 1 for all elements.

5 Regularization-Based Enhancement

In this section, an MSE-based regularization of our approach is described. And we attempt to combine a small amount of relevant labels to explore more possibilities of our proposal.

5.1 An MSE-Based Regularization

Previous works indicate that CE loss always makes the model focus on hard instances that are difficult to be classified precisely, while MSE loss and Mean Absolute Error (MAE) loss are less sensitive to hard instances since they treat per instance coequally [18], [49]. As this property, the convergence rate of CE loss is superior to MSE loss and MAE loss, whereas this property makes CE loss more prone to the overfitting problem than MSE loss and MAE loss when noisy labels present at training data [18], [49]. Actually, an excellent approach can converge quickly during the training process, and shows good generalization ability and robustness for unseen instances [16].

Obviously, BCE loss has a similar property to CE loss, which results in an excellent convergence rate of approaches. Meanwhile, approaches based on BCE loss are easy to suffer from the overfitting problem when using noisy labeled data to learn. In fact, ML-CLL is a problem setting with dense noisy labels, BCE loss may cause the overfitting problem of a model in ML-CLL. To cope with this problem, we introduce an MSE-based regularizer based on MSE loss (i.e. ℓ_2-norm regularization) to balance the robustness and convergence requirement of the proposed approach. Hence, the MSE-based regularizer is defined as:

$$\bar{L}_{mse}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) = \left\| \bar{\boldsymbol{y}} - \mathbf{T}^T\boldsymbol{f}(\boldsymbol{x}) \right\|_F^2. \tag{10}$$

Finally, we combine the complementary loss and the MSE-based regularizer term, which leads to our target loss:

$$\bar{\mathcal{L}}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) = \bar{L}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) + \beta\bar{L}_{mse}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}), \tag{11}$$

where β is the trade-off parameter and set as 1 (the selection shown in Section 6). The all procedure of the proposed approach (called MLCL) is shown in Algorithm 1.

5.2 Incorporation of Relevant Labels

In many practical situations, we can use complementary labels and relevant labels to learn more accurate classifiers, which is highly practical implementation. To this end, motivated by [11], [50], let us design a reasonable combination of the loss derived from complementary labeled data and relevant labeled data:

$$\tilde{\mathcal{L}}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}, \tilde{\boldsymbol{y}}) = \bar{L}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) + \|\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x})\|_F^2, \tag{12}$$

TABLE 2
Statistics of datasets.

Datasets	$\lvert\mathcal{S}\rvert$	$dim(\mathcal{S})$	$L(\mathcal{S})$	$LCard(\mathcal{S})$
scene	2407	294	6	1.07
yeast	2417	103	14	4.23
eurlex_dc	8636	5000	15	1.02
eurlex_sm	13270	5000	15	1.74
corel5k	4194	499	15	1.70
corel16k	11103	120	15	1.77
bookmark	38912	2150	15	1.25
delicious	14784	500	15	4.32

where $\tilde{\boldsymbol{y}} = [\tilde{y}^1, \ldots, \tilde{y}^1] \in \{0,1\}^K$ denotes a binary vector of relevant labels $\tilde{Y}$ of $\boldsymbol{x}$, in which $\tilde{y}^1 = 1$ when the label $l_k \in \tilde{Y}$. To provide more practicability, we do not restrict given relevant labels $\tilde{Y}$ to must be equal to the set of relevant labels Y, which means $\tilde{Y} \subseteq Y$ and $\tilde{Y} \neq \emptyset$.

As explained in the instruction, we can naturally collect data associated with complementary labels and relevant labels via crowdsourcing [19]. Our loss function Eq. (12) can leverage both kinds of labeled data to learn better classifiers. We will experimentally show the usefulness of this combination method in Section 6.

6 Experiments

In this section, we will evaluate the effectiveness of MLCL, where five common MLL criteria, including *ranking loss*, *hamming loss*, *one error*, *coverage* and *average precision*, are employed in this paper. Smaller values for the first four criteria indicate better performance, while a higher value of *average precision* indicates better performance. The label set of $\boldsymbol{x}$ is predicted by $Y = \{l_k | f^k(\boldsymbol{x}) > 0.5, 1 \leq k \leq K\}$. All experiments were conducted using PyTorch [51] and NVIDIA TESLA K80 GPU to implement. The code will be released after this paper has been accepted.

6.1 Experimental Settings

Datasets. We use eight widely-used MLL datasets, namely *corel5k*, *corel16k*, *delicious*, *eurlex_dc*, *eurlex_sm*, *yeast*, *bookmarks* and *scene*, to our experiments[2]. Following [40], [41], we adopt the same pre-processing to deal with the datasets. More specifically, rare class labels are filtered out for datasets with more than 15 class labels, whose class labels are kept under 15. Accordingly, instances that are relevant with removed class labels are filtered out as well. Detailed characteristics of these datasets are shown in Table 2.

Base models. The linear model is used as the base model.

Baselines. Two typical MLL approaches, ML-KNN [26] and LIFT [52], are utilized as baselines, which deal with ML-CLL via regarding all possible labels in the candidate label set as relevant labels for a training instance. Similarly, three recent PML approaches are employed as comparing approaches, including PML-lc [40], fpml [43] and PML-LRS [42], which learn from training instances associated with candidate labels. In addition, we employ a multi-class CLL approach, called L-UW [15], as a baseline, which uses BEC loss and *sigmoid* output layer instead of CE loss and *softmax* output layer respectively to make L-UW suit for multi-labeled data.

2. Publicly available at http://mulan.sourceforge.net/datasets.

TABLE 3
Experimental results (mean ± std) on training data with uniform complementary labels. The best performance of each dataset is presented in **boldface**, where •/○ indicates whether MLCL is superior/inferior to baselines (with 5% t-test).

Methods	ML-KNN	LIFT	fpml	PML-lc	PML-LRS	L-UW	MLCL
				Ranking loss ↓			
scene	.340±.032•	.289±.020•	.504±.025•	.490±.025•	**.258±.007**○	.372±.028•	.259±.030
yeast	.247±.012•	.298±.012•	.233±.013•	.251±.015•	.464±.019•	.214±.011	**.211±.013**
eurlex_dc	.303±.016•	.286±.016•	.488±.033•	.347±.025•	.316±.011•	.598±.024•	**.229±.026**
eurlex_sm	.336±.010•	.346±.012•	.488±.006•	.436±.011•	.332±.009•	.646±.015•	**.312±.014**
corel5k	.379±.034	.433±.037•	.444±.026•	.406±.075•	**.334±.009**○	.367±.031	.349±.035
corel16k	.328±.047	.392±.027•	.420±.033•	.457±.046•	.303±.005	.303±.035	**.289±.042**
bookmark	.384±.006•	.310±.007•	.469±.019•	.454±.036•	.260±.004	.303±.010•	**.252±.013**
delicious	.398±.004•	.383±.003•	.438±.008•	.445±.015•	.305±.002	.302±.006	.310±.003
				One Error ↓			
scene	.692±.030•	.605±.023•	.815±.027•	.717±.021•	.540±.023•	.609±.041•	**.427±.018**
yeast	.297±.029•	.284±.028•	.251±.025	.583±.026•	.738±.102•	.251±.025	**.251±.023**
eurlex_dc	.776±.031•	.670±.013•	.925±.016•	.774±.015•	.847±.010•	.837±.034•	**.594±.035**
eurlex_sm	.689±.012•	.679±.009•	.872±.011•	.662±.012	.731±.005•	.696±.008•	**.656±.029**
corel5k	.815±.048•	.842±.056•	.854±.035•	.811±.062•	.756±.010	.769±.034	**.736±.065**
corel16k	.736±.056	.789±.046•	.816±.025•	.946±.028•	.730±.000•	.693±.057	**.690±.056**
bookmark	.801±.006•	.649±.016•	.885±.020•	.798±.005•	.584±.005•	.590±.022•	**.509±.012**
delicious	.592±.018•	.533±.015•	.618±.017•	.679±.011•	.452±.007	.467±.023•	**.448±.016**
				Hamming loss ↓			
scene	.820±.002•	.820±.003•	.819±.002•	**.251±.007**	.814±.000•	.518±.042•	.264±.027
yeast	.697±.012•	.697±.013•	.697±.013•	.268±.010•	.316±.000•	.243±.010	**.235±.008**
eurlex_dc	.932±.000•	.932±.000•	.118±.006•	.104±.002•	.890±.039•	.806±.015•	**.092±.005**
eurlex_sm	.883±.001•	.883±.001•	.148±.005•	**.138±.002**	.825±.027•	.773±.008•	.139±.005
corel5k	.886±.007•	.887±.007•	.887±.007•	**.155±.004**	.869±.002•	.463±.018•	.229±.068
corel16k	.882±.009•	.882±.009•	.882±.009•	**.177±.011**○	.862±.001•	.423±.033•	.202±.067
bookmark	.917±.001•	.916±.001•	.420±.009•	**.123±.001**○	.813±.001•	.409±.014•	.140±.004
delicious	.711±.003•	.711±.003•	.711±.003•	.394±.011•	.459±.002•	.369±.027•	**.289±.004**
				Coverage ↓			
scene	.299±.026•	.256±.017•	.434±.021•	.420±.021•	**.230±.006**	.328±.022•	.234±.025
yeast	.579±.018•	.649±.020•	.553±.033•	**.506±.023**	.742±.027•	.525±.017	.525±.021
eurlex_dc	.285±.014•	.269±.015•	.458±.031•	.326±.023•	.298±.010•	.334±.017•	**.204±.023**
eurlex_sm	.416±.010•	.427±.013•	.569±.010•	.509±.013•	.419±.010•	.519±.008•	**.365±.014**
corel5k	.473±.034	.516±.035•	.529±.028•	.492±.072	**.429±.008**	.457±.038	.445±.048
corel16k	.430±.044	.488±.027•	.513±.035•	.537±.051•	.405±.008	.407±.033	**.393±.042**
bookmark	.359±.007•	.328±.008•	.475±.019•	.458±.035•	.280±.004	.292±.011•	**.279±.011**
delicious	.712±.006•	.703±.004•	.726±.009•	.695±.009•	**.609±.003**○	.613±.006○	.632±.007
				Average Precision ↑			
scene	.543±.024•	.600±.017•	.417±.021•	.465±.018•	.637±.011•	.568±.026•	**.699±.017**
yeast	.677±.019•	.636±.017•	.688±.017•	.610±.016•	.459±.032•	.712±.020	**.718±.019**
eurlex_dc	.412±.018•	.471±.012•	.232±.022•	.373±.015•	.346±.009•	.250±.031•	**.549±.025**
eurlex_sm	.419±.010•	.421±.010•	.273±.006•	.367±.009•	.402±.005•	.285±.009•	**.474±.017**
corel5k	.355±.035•	.307±.038•	.297±.023•	.330±.044•	**.397±.010**	.371±.028	.391±.037
corel16k	.405±.050	.350±.035•	.325±.022•	.248±.026•	.424±.006	.437±.044	**.449±.049**
bookmark	.383±.007•	.480±.010•	.267±.019•	.329±.016•	.534±.004•	.506±.014•	**.584±.013**
delicious	.487±.006•	.511±.004•	.457±.006•	.446±.010•	**.580±.002**	.570±.009	.572±.005

6.2 Comparison on Uniform Complementary Labels

Setup. Weight-decay is set as $1e-4$ and learning rate is selected from $\{1e-1, 1e-2, 1e-3\}$ for all data sets. We employ Adam [53] optimization method, and set the number of batch-size and epoch as 256 and 200 respectively. L-UW applies the same model and hyper-parameters as ours. Here, we estimate $\mathbf{T}$ with a linear model. We use Ten-fold cross-validation to evaluate experiments, where training data is associated with complementary labels that are generated by randomly selecting one of possible labels excepting relevant labels (uniform complementary labels), and test data is equipped with the set of relevant labels. The mean metrics value and *standard deviation* (std) will be reported as final experimental results for all approaches.

Results. Table 3 is utilized to report experimental results of various approaches on eight data sets equipped with uniform complementary labels. ↑ / ↓ indicates the larger/smaller the value, the better the performance.

According to reported results in Table 3, we can observe that results of MLCL are superior or comparable performance against baselines out of different data sets on five criteria. Our approach achieves the best performance in most cases. Specifically, the proposed approach outperforms LIFT on eight datasets across all metrics. This is because

TABLE 4
Experimental results (mean ± std) on training data with biased complementary labels. The best performance of each dataset is presented in **boldface**, where •/○ represents whether MLCL is superior/inferior to baselines (with 5% t-test).

Methods	ML-KNN	LIFT	fpml	PML-lc	PML-LRS	L-UW	MLCL
				Ranking loss↓			
scene	**.086±.015**○	.319±.025	.486±.027•	.492±.019•	.258±.013○	.368±.025•	.326±.050
yeast	.240±.014•	.297±.016•	.227±.013•	.248±.012•	.454±.024•	.202±.012	**.199±.012**
eurlex_dc	.668±.009•	.636±.021•	.537±.015•	.349±.028•	.326±.009	.586±.036•	**.308±.034**
eurlex_sm	.364±.020•	.392±.014•	.499±.019•	.447±.012•	.333±.009•	.641±.015•	**.316±.016**
corel5k	**.324±.038**○	.431±.030•	.474±.028•	.386±.047	.357±.012	.382±.033	.358±.039
corel16k	.413±.063•	.431±.041•	.454±.033•	.471±.068•	.375±.015	.373±.029	**.357±.040**
bookmark	.567±.007•	.449±.042•	.552±.018•	.491±.016•	.244±.003•	.326±.008•	**.211±.011**
delicious	.430±.005•	.413±.005•	.452±.008•	.433±.011•	**.314±.003**○	.349±.012○	.360±.008
				One Error↓			
scene	**.228±.032**○	.669±.043•	.803±.038•	.720±.018•	.613±.017•	.696±.025•	.553±.054
yeast	.330±.032•	.280±.025•	.254±.028	.583±.027•	.546±.097•	.256±.025	**.254±.024**
eurlex_dc	.977±.005•	.959±.014•	.947±.008•	.774±.015•	.822±.004•	.822±.038•	**.695±.074**
eurlex_sm	.699±.016•	.753±.036•	.886±.024•	.664±.014	.737±.011•	.704±.012•	**.650±.045**
corel5k	**.738±.067**	.851±.038•	.861±.034•	.828±.059•	.747±.016	.792±.039•	.752±.037
corel16k	.780±.061•	.827±.049•	.837±.025•	.952±.021•	.730±.000	.731±.053	**.707±.063**
bookmark	.906±.007•	.804±.037•	.925±.008•	.792±.004•	.576±.003•	.635±.022•	**.502±.008**
delicious	.585±.012•	.557±.013•	.617±.025•	.681±.012•	**.434±.006**○	.485±.016•	.463±.017
				Hamming loss ↓			
scene	**.088±.009**○	.819±.002•	.820±.002•	**.252±.006**	.814±.000•	.523±.048•	.290±.029
yeast	.697±.012•	.697±.013•	.697±.013•	.268±.010•	.316±.000•	.253±.017•	**.239±.008**
eurlex_dc	.932±.000•	.932±.000•	.118±.007•	**.104±.002**	.889±.039•	.799±.035•	.109±.011
eurlex_sm	.883±.001•	.883±.001•	.148±.005•	.139±.002	.825±.027•	.772±.009•	**.138±.007**
corel5k	**.114±.008**○	.887±.007•	.887±.007•	.157±.003•	.869±.002•	.498±.012•	.208±.033
corel16k	.882±.009•	.882±.009•	.882±.009•	**.178±.010**	.862±.001•	.481±.028•	.207±.086
bookmark	.917±.001•	.916±.001•	.419±.009•	**.122±.001**○	.813±.003•	.549±.046•	.146±.003
delicious	.711±.003•	.711±.003•	.711±.003•	.388±.013•	.459±.002•	.453±.015•	**.304±.005**
				Coverage↓			
scene	**.086±.013**○	.280±.020	.420±.023•	.420±.016•	.229±.011○	.321±.021•	.286±.041
yeast	.551±.017•	.638±.028•	.533±.012•	**.493±.025**	.723±.040•	.500±.018	.498±.021
eurlex_dc	.626±.008•	.596±.019•	.504±.014•	.328±.026•	.306±.009•	.333±.018•	**.274±.030**
eurlex_sm	.432±.018•	.456±.014•	.579±.015•	.520±.015•	.418±.009•	.512±.009•	**.362±.016**
corel5k	**.419±.055**	.515±.024•	.555±.031•	.480±.041	.451±.013	.470±.036	.449±.038
corel16k	.498±.052•	.521±.038•	.542±.035•	.533±.066•	.454±.018	.468±.030	**.453±.039**
bookmark	.565±.006•	.455±.039•	.553±.017•	.492±.014•	.265±.003•	.308±.013•	**.231±.011**
delicious	.736±.004•	.723±.005•	.737±.008•	.691±.009	**.625±.003**○	.671±.012○	.688±.006
				Average Precision ↑			
scene	**.860±.020**○	.559±.028•	.428±.026•	.462±.014•	.608±.013	.529±.020•	.618±.046
yeast	.670±.023•	.634±.016•	.691±.022•	.614±.015•	.500±.026•	.719±.020	**.726±.018**
eurlex_dc	.145±.005•	.166±.016•	.201±.009•	.371±.020•	.357±.005•	.266±.031•	**.456±.061**
eurlex_sm	.405±.013•	.373±.016•	.262±.016•	.366±.010•	.400±.007•	.282±.011•	**.482±.025**
corel5k	**.409±.040**	.300±.030•	.282±.017•	.325±.048•	.392±.017	.352±.032	.380±.037
corel16k	.355±.054•	.318±.033•	.301±.024•	.240±.030•	.393±.054	.384±.036	**.407±.047**
bookmark	.219±.004•	.320±.037•	.212±.007•	.320±.004•	.544±.003•	.469±.014•	**.599±.008**
delicious	.473±.006•	.490±.006•	.450±.008•	.449±.010•	**.581±.002**○	.544±.010	.544±.009

our approach is better at tackling the issue that training data is associated with relevant labels and irrelevant labels simultaneously than fully supervised MLL algorithms. Furthermore, experimental results of PML-lc and PML-LRS are inferior to ours in most cases, which demonstrates that PML approaches are indeed inferior to our approach in cases of dense noisy labels. Similarly, based on the results of L-UW shown in Table 3, we observe that our approach outperforms L-UW on almost all datasets and metrics other than *ranking loss* and *coverage* on the delicious dataset. This reflects that label correlations are important to solve ML-CLL problems, which leads to the proposed approach taking label correlations into account surpasses L-UW that ignores label correlations.

6.3 Comparison on Biased Complementary Labels

Setup. To evaluate the effectiveness of our approach in different situations, we utilize training data with biased complementary labels that are generated via the co-occurrence rate of relevant labels. Specifically, we select a complementary label of an instance $\boldsymbol{x}$ from $\mathcal{Y} \setminus Y$, and the selecting rule follows: the class label with a lower co-occurrence rate has a higher probability to be selected as a complementary label. We adopt training data with biased complementary labels

TABLE 5
Ablation experimental results (mean ± std) on training data with uniform complementary labels. The best performance is in **boldface**.

Methods	Uniform complementary labels				Biased complementary labels			
	scene	yeast	eurlex_dc	corel5k	scene	yeast	eurlex_dc	corel5k
	Hamming loss↓							
MLCL	**.264±.027**	.235±.008	**.092±.005**	**.229±.068**	**.290±.029**	.239±.008	.109±.011	**.208±.033**
Without $\mathbf{C}$	.290±.039	.421±.011	.109±.018	.466±.025	.294±.029	.409±.012	**.088±.004**	.444±.031
Without $\bar{L}_{mse}$	.510±.044	**.229±.007**	.509±.043	.461±.053	.481±.047	**.230±.009**	.512±.046	.489±.036
	Ranking loss↓							
MLCL	**.259±.030**	**.211±.013**	**.229±.026**	**.349±.035**	**.326±.050**	**.199±.012**	.308±.034	**.358±.039**
Without $\mathbf{C}$	.282±.063	.419±.018	.277±.041	.487±.021	.348±.046	.406±.016	**.268±.024**	.467±.026
Without $\bar{L}_{mse}$	.379±.024	.216±.010	.303±.028	.362±.030	.353±.018	.204±.011	.320±.025	.387±.027
	One error↓							
MLCL	**.427±.018**	.251±.023	**.594±.035**	.736±.065	**.553±.054**	**.254±.024**	.695±.074	**.752±.037**
Without $\mathbf{C}$	.474±.047	.633±.043	.708±.106	.866±.019	.560±.042	.612±.051	**.564±.029**	.855±.027
Without $\bar{L}_{mse}$	.607±.037	**.250±.025**	.740±.048	**.734±.058**	.686±.013	.256±.025	.753±.044	.773±.068
	Coverage↓							
MLCL	**.234±.025**	**.525±.021**	**.204±.023**	**.445±.048**	**.286±.041**	**.498±.021**	.274±.030	**.449±.038**
Without $\mathbf{C}$	.255±.055	.683±.029	.247±.035	.565±.032	.306±.039	.660±.023	**.240±.023**	.547±.031
Without $\bar{L}_{mse}$	.334±.020	.527±.011	.249±.024	.451±.035	.310±.015	.501±.015	.265±.022	.473±.023
	Average precision↑							
MLCL	**.699±.017**	**.718±.019**	**.549±.025**	**.391±.037**	**.618±.046**	**.726±.018**	**.456±.061**	**.380±.037**
Without $\mathbf{C}$	.671±.045	.472±.018	.469±.085	.274±.014	.611±.038	.489±.015	.447±.021	.289±.022
Without $\bar{L}_{mse}$	.566±.023	.711±.019	.426±.040	.389±.041	.541±.013	.717±.020	.411±.034	.359±.050

TABLE 6
Parameter sensitivity analysis on uniform complementary-label data, where metric is *average precision*. The best performance is in **boldface**.

β	scene	yeast	eurlex_dc	eurlex_sm	corel5k	corel16k	bookmark	delicious
0.1	.678±.017	.714±.019	.545±.019	.451±.025	.374±.033	.444±.046	.565±.007	.554±.005
0.3	.683±.015	.716±.018	.549±.021	.460±.021	.378±.032	.447±.047	.579±.011	.565±.005
0.5	.687±.016	.718±.018	.547±.022	.463±.016	.385±.031	.447±.048	.583±.008	**.575±.005**
0.8	.693±.016	.718±.018	.541±.022	.469±.018	.387±.037	.448±.048	.582±.007	.572±.006
1	**.699±.017**	**.718±.019**	**.549±.025**	**.474±.017**	**.391±.037**	**.449±.049**	**.584±.013**	.572±.005

to train the model, while test data is equipped with relevant label sets to evaluate the effectiveness of our approach. For other experimental settings, we apply same settings with Subsection 5.2.

Results. The mean and std of results on test data are shown in Table 4. According to results shown in Table 4, we can summarize the following impressive observations: (1) MLCL achieves superior or comparable performance to LIFT, fpml, PML-lc, PML-LRS and L-UW on different data sets, which proves that the proposed approach can predict the set of proper labels for unseen instances from complementary labeled data; (2) Although MLCL fails to achieve the best result on the scene dataset, our approach is better than other baselines in the rest of datasets, which indicates that our approach can effectively deal with ML-CLL problems than others. These observations demonstrate that the proposed method can both hold for the situation of data with uniform and biased complementary labels.

6.4 Additional Experiments

Ablation experiments. We then explore the effect of different learning components on MLCL performance. Table 5 summarizes results of MLCL without the different components, which are trained on the data with uniform complementary labels. In Table 5, without $\mathbf{C}$ refers to MLCL directly using the estimated initial transition matrix $\mathbf{S}$ for training, and without $\bar{L}_{mse}$ indicates that MLCL only utilizes Eq. (9) for optimization.

From results reported in Table 5, the performance of MLCL surpasses that without different components in most cases, which shows that two components, including using label correlations to correct and an MSE-based regularizer, are beneficial for our approach to improve the performance. Especially, estimating $\mathbf{T}$ based on label correlations pushes the proposed approach performance forward significantly compared with that without $\mathbf{C}$ on most cases. Similarly, an MSE-based regularizer brings significant benefits for our approach, which demonstrates that an MSE-based regularizer balances the robustness and convergence rate of BCE loss. These indicate that using label correlations to estimate the transition matrix $\mathbf{T}$ and an MSE-based regularizer are effective strategies to alleviate ML-CLL problems.

Trade-off parameter β. Table 6 reports the performance of MLCL with varying β values that trade-off the complementary loss function $\bar{L}$ and an MSE-based regularization $\bar{L}_{mse}$. Here, *average precision* is regarded as the criterion, and the training data is with uniform complementary labels. β is selected from the candidate value list $\{0.1, 0.3, 0.5, 0.8, 1\}$. We can observe the best results of most datasets is achieved at $\beta = 1$ and the performance drops when β takes a smaller value. In general, a relatively large β ($\beta \leq 1$) usually leads to better performance than a small value. Therefore, we set

TABLE 7
Experimental results (mean ± std) of five criteria."Fully supervised" is the linear model training with the fully supervised data (fully supervised MLL). "CL" denotes that each instance is associated with **a complementary label** sampled uniformly. "RL" presents that the model only using **relevant labels** $\tilde{Y}$ to train. "CL & RL" uses the linear model with the loss function Eq. (12) to train, where each instance is equipped with **a complementary label** and **a relevant label**.

Datasets	scene	yeast	eurlex_dc	eurlex_sm	corel5k	corel16k	bookmark	delicious
				Hamming loss↓				
Fully supervised	.120±.013	.208±.009	.004±.000	.033±.001	.198±.012	.196±.012	.098±.004	.276±.006
CL	.264±.027	.235±.008	.092±.005	.139±.005	.229±.068	.202±.067	.140±.004	.289±.004
RL	.128±.012	.241±.011	.005±.001	.055±.005	.184±.014	.152±.010	.076±.002	.285±.004
CL & RL	.124±.008	.225±.010	.005±.001	.053±.002	.178±.012	.172±.010	.085±.002	.285±.004
				Ranking loss↓				
Fully supervised	.075±.009	.169±.009	.003±.001	.019±.001	.258±.029	.222±.029	.090±.005	.226±.004
CL	.259±.030	.211±.013	.229±.026	.312±.014	.349±.035	.289±.042	.252±.013	.310±.003
RL	.081±.009	.23±.013	.005±.001	.052±.027	.269±.031	.250±.025	.116±.006	.295±.008
CL & RL	.082±.011	.191±.011	.005±.001	.044±.002	.268±.031	.227±.021	.102±.004	.267±.004
				One Error↓				
Fully supervised	.222±.032	.223±.023	.019±.004	.069±.005	.627±.038	.588±.056	.313±.009	.340±.012
CL	.427±.018	.251±.023	.594±.035	.656±.029	.736±.065	.690±.056	.509±.012	.448±.016
RL	.231±.032	.280±.024	.022±.005	.106±.02	.641±.042	.616±.042	.338±.007	.438±.019
CL & RL	.229±.033	.255±.032	.022±.005	.098±.007	.639±.040	.600±.044	.324±.007	.398±.017
				Coverage↓				
Fully supervised	.077±.009	.451±.019	.004±.000	.074±.002	.347±.044	.315±.024	.112±.005	.527±.007
CL	.234±.025	.525±.021	.204±.023	.365±.014	.445±.048	.393±.042	.279±.011	.632±.007
RL	.083±.008	.561±.024	.007±.002	.123±.035	.364±.048	.353±.025	.142±.006	.582±.008
CL & RL	.084±.010	.474±.021	.006±.001	.113±.004	.363±.048	.326±.020	.125±.004	.564±.006
				Average Precision↑				
Fully supervised	.868±.018	.760±.015	.988±.003	.943±.004	.494±.024	.530±.038	.766±.007	.662±.005
CL	.699±.017	.718±.019	.549±.025	.474±.017	.391±.037	.449±.049	.584±.013	.572±.005
RL	.860±.018	.704±.013	.984±.003	.888±.032	.484±.028	.505±.028	.737±.006	.590±.009
CL & RL	.860±.019	.734±.018	.985±.004	.899±.004	.485±.028	.523±.030	.753±.006	.618±.005

$\beta = 1$ for MLCL.

6.5 Combination of Complementary Labels and Relevant Labels

Setup. Finally, we demonstrate the effectiveness of combining relevant labeled data and complementary labeled one. The training data is associated with uniform complementary labels and relevant labels simultaneously. More specifically, an instance $\boldsymbol{x}$ is associated with a complementary label $\bar{y}$ and relevant labels $\tilde{Y}$, where $\bar{y}$ is uniformly selected and $\tilde{Y}$ is randomly selected from the relevant label set Y of $\boldsymbol{x}$ (i.e., $\tilde{Y} \subseteq Y$). Here, we set $|\tilde{Y}| = 1$ that means each instance only associated with a complementary label and a relevant label. The other experimental settings are the same with Subsection 5.2.

Results. We compare three methods: (1) the "Fully supervised" method uses the linear model to train with the fully supervised data, which is fully supervised MLL; (2) the "CL" method refers to MLCL training with the uniform complementary-label data; (3) the "RL" method uses the linear model to train with relevant labels $\tilde{Y}$; (4) the combination ("CL & RL") method adopts the linear model with the loss function Eq. (12) to train, where the training data is equipped with a combination of complementary labels and relevant labels. Table 7 reports the experimental results on five criteria. We can see that the performance of "CL& RL" method is much superior to "CL" method on all datasets over *hamming loss*, *ranking loss*, *one error*, *coverage* and *average precision*, such as "CL& RL" method outperforms "CL" method by a large margin over *average precision* (**+0.436** on eurlex_dc and **+0.425** on eurlex_sm). This demonstrates that the ML-CLL is easily applied to fully supervised MLL scenarios, MLL with missing labels [54], [55] or other MLL scenarios. Moreover, the results in "CL & RL" achieve comparable performance to "RL" method, which illustrates that using complementary labels improves performance and the information encompassed by relevant labels does not overshadow or encompass the information from complementary labels. Finally, the results of "CL & RL" method close to "Fully supervised" method, which illustrates that ML-CLL can get excellent results with just a small amount of additional information. This is useful for application in the real world because ML-CLL can obtain good performance through less expensive labeled data.

7 Conclusion

In this paper, we theoretically analyze the reason why the estimated transition matrix in multi-class CLL is distorted in ML-CLL. To alleviate the difficulty in directly calculating the transition matrix from complex label correlations under multi-labeled data, we propose a two-step method to estimate the transition matrix $\mathbf{T}$ in ML-CLL, which uses label correlations to correct an initial transition matrix. Furthermore, we theoretically show that the proposed approach is classifier-consistent. Additionally, due to the robustness of MSE loss, an MSE-based regularizer is introduced to alleviate the tendency of the fast convergent BCE loss overfitting to noises. Finally, we show that our proposed ML-

CLL can be easily combined with relevant labels and the proposed method can achieve performance comparable to fully supervised MLL with just a small amount of additional information.

ACKNOWLEDGMENTS

The authors wish to thank the anonymous reviewers for their helpful comments and suggestions. This work was supported by the National Science Foundation of China (62176055). MX is supported by the Australian Research Council (DE230101116). 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 Trans. Knowl. Data Eng.*, vol. 26, no. 8, pp. 1819–1837, 2014.

[2] M.-L. Zhang and L. Wu, "Lift: Multi-label learning with label-specific features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 1, pp. 107–120, 2015.

[3] W. Tang, W. Zhang, and M.-L. Zhang, "Multi-instance partial-label learning: Towards exploiting dual inexact supervision," *Science China Information Sciences*, vol. 67, no. 3, pp. 1–14, 2024.

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

[5] P.-J. Tang, M. Jiang, B. N. Xia, J. W. Pitera, J. Welser, and N. V. Chawla, "Multi-label patent categorization with non-local attention-based graph convolutional network," in *Proceedings of the 34th Conference on Artificial Intelligence*, York, NY, 2020, pp. 9024–9031.

[6] A. Lambrecht and C. Tucker, "When does retargeting work? information specificity in online advertising," *Journal of Marketing research*, vol. 50, no. 5, pp. 561–576, 2013.

[7] Y. Yang, J. Guo, G. Li, L. Li, W. Li, and J. Yang, "Alignment efficient image-sentence retrieval considering transferable cross-modal representation learning," *Frontiers of Computer Science*, vol. 18, no. 1, p. 181335, 2024.

[8] W. Zhang, F. Liu, L. Luo, and J. Zhang, "Predicting drug side effects by multi-label learning and ensemble learning," *BMC bioinformatics*, vol. 16, pp. 1–11, 2015.

[9] Y. Qiu, F. Lin, W. Chen, and M. Xu, "Pre-training in medical data: A survey," *Machine Intelligence Research*, vol. 20, no. 2, pp. 147–179, 2023.

[10] Y. Gao, M. Xu, and M.-L. Zhang, "Unbiased risk estimator to multi-labeled complementary label learning," in *Proceedings of the 32nd International Joint Conference on Artificial Intelligence*, Macao, China, 2023, pp. 3732–3740.

[11] T. Ishida, G. Niu, W.-H. Hu, and M. Sugiyama, "Learning from complementary labels," in *Advances in Neural Information Processing Systems 30*, Long Beach, CA, 2017, pp. 5639–5649.

[12] T. Ishida, G. Niu, A. K. Menon, and M. Sugiyama, "Complementary-label learning for arbitrary losses and models," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, Long Beach, CA, 2019, pp. 2971–2980.

[13] X.-Y. Yu, T.-L. Liu, M.-M. Gong, and D.-C. Tao, "Learning with biased complementary labels," in *Proceedings of the 15th European Conference on Computer Vision*, Munich, Germany, 2018, pp. 69–85.

[14] Y.-T. Chou, G. Niu, H.-T. Lin, and M. Sugiyama, "Unbiased risk estimators can mislead: A case study of learning with complementary labels," in *Proceedings of the 37th International Conference on Machine Learning*, Virtual Event, 2020, pp. 1929–1938.

[15] Y. Gao and M.-L. Zhang, "Discriminative complementary-label learning with weighted loss," in *Proceedings of the 38th International Conference on Machine Learning*, Virtual Event, 2021, pp. 3587–3597.

[16] D.-B. Wang, L. Feng, and M.-L. Zhang, "Learning from complementary labels via partial-output consistency regularization," in *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, Virtual Event, 2021, pp. 3075–3081.

[17] L. Feng, T. Kaneko, B. Han, G. Niu, B. An, and M. Sugiyama, "Learning with multiple complementary labels," in *Proceedings of the 37th International Conference on Machine Learning*, Virtual Event, 2020, pp. 3072–3081.

[18] A. Ghosh, H. Kumar, and P. S. Sastry, "Robust loss functions under label noise for deep neural networks," in *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, San Francisco, CA, 2017, pp. 1919–1925.

[19] T. S. Sindlinger, "Crowdsourcing: why the power of the crowd is driving the future of business," 2010.

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

[21] F. Wu, Z.-H. Wang, Z.-F. Zhang, Y. Yang, J.-B. Luo, W.-W. Zhu, and Y.-T. Zhuang, "Weakly semi-supervised deep learning for multi-label image annotation," *IEEE Trans. Big Data*, vol. 1, no. 3, pp. 109–122, 2015.

[22] S. S. Bucak, R. Jin, and A. K. Jain, "Multi-label learning with incomplete class assignments," in *The 24th IEEE Conference on Computer Vision and Pattern Recognition*, Colorado Springs, CO, 2011, pp. 2801–2808.

[23] W.-W. Liu, I. W. Tsang, and K. Müller, "An easy-to-hard learning paradigm for multiple classes and multiple labels," *J. Mach. Learn. Res.*, vol. 18, pp. 94:1–94:38, 2017.

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

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

[26] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," *Pattern Recognit.*, vol. 40, no. 7, pp. 2038–2048, 2007.

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

[28] J. Fürnkranz, E. Hüllermeier, E. Loza Mencía, and K. Brinker, "Multilabel classification via calibrated label ranking," *Machine learning*, vol. 73, no. 2, pp. 133–153, 2008.

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

[30] G. Tsoumakas, I. Katakis, and I. P. Vlahavas, "Random k-labelsets for multilabel classification," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 7, pp. 1079–1089, 2011.

[31] M.-L. Zhang and Z.-H. Zhou, "Multilabel 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.

[32] Y.-C. Li, Y. Song, and J.-B. Luo, "Improving pairwise ranking for multi-label image classification," in *Proceedings of 2017 IEEE conference on computer vision and pattern recognition*, Honolulu, HI, 2017, pp. 3617–3625.

[33] S.-W. Ji, L. Tang, S.-P. Yu, and J.-P. Ye, "A shared-subspace learning framework for multi-label classification," *ACM Trans. Knowl. Discov. Data*, vol. 4, no. 2, pp. 8:1–8:29, 2010.

[34] W. Gerych, T. Hartvigsen, L. Buquicchio, E. Agu, and E. A. Rundensteiner, "Recurrent bayesian classifier chains for exact multi-label classification," in *Advances in Neural Information Processing Systems 34*, virtual event, 2021, pp. 15 981–15 992.

[35] W.-T. Zhao, S.-F. Kong, J.-W. Bai, D. Fink, and C. P. Gomes, "HOT-VAE: learning high-order label correlation for multi-label classification via attention-based variational autoencoders," in *Proceedings of 35th AAAI Conference on Artificial Intelligence*, Virtual Event, 2021, pp. 15 016–15 024.

[36] L.-C. Wang, Z.-M. Ding, S.-J. Han, J.-J. Han, C. Choi, and Y. Fu, "Generative correlation discovery network for multi-label learning," in *Proceedings of 2019 IEEE International Conference on Data Mining*, Beijing, China, 2019, pp. 588–597.

[37] G.-X. Xun, K. Jha, J.-H. Sun, and A.-D. Zhang, "Correlation networks for extreme multi-label text classification," in *Proceedings of 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Virtual Event, 2020, pp. 1074–1082.

[38] L. Sun, S. Feng, J. Liu, G. Lyu, and C. Lang, "Global-local label correlation for partial multi-label learning," *IEEE Transactions on Multimedia*, vol. PP, no. 99, pp. 1–1, 2021.

[39] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *National science review*, vol. 5, no. 1, pp. 44–53, 2018.

[40] M.-K. Xie and S.-J. Huang, "Partial multi-label learning," in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, New Orleans, LA, 2018, pp. 4302–4309.

[41] S.-J. H. Ming-Kun Xie, "Partial multi-label learning with noisy label identification," in *Proceedings of 34th AAAI Conference on Artificial Intelligence*, New York, NY, 2020, pp. 6454–6461.

[42] L.-J. Sun, S.-H. Feng, T. Wang, C.-Y. Lang, and Y. Jin, "Partial multi-label learning by low-rank and sparse decomposition," in *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, Honolulu, HI, 2019, pp. 5016–5023.

[43] G.-X. Yu, X. Chen, C. Domeniconi, J. Wang, Z. Li, Z.-L. Zhang, and X.-D. Wu, "Feature-induced partial multi-label learning," in *Proceedings of 2018 IEEE International Conference on Data Mining*, Singapore, 2018, pp. 1398–1403.

[44] Y.-W. Xu, M.-M. Gong, J.-X. Chen, T.-L. Liu, K. Zhang, and K. Batmanghelich, "Generative-discriminative complementary learning," in *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, New York, NY, 2020, pp. 6526–6533.

[45] S. Diplaris, G. Tsoumakas, P. A. Mitkas, and I. P. Vlahavas, "Protein classification with multiple algorithms," in *Advances in 10th Panhellenic Conference on Informatics*, vol. 3746, Volos, Greece, 2005, pp. 448–456.

[46] G. Patrini, A. Rozza, A. K. Menon, R. Nock, and L.-Z. Qu, "Making deep neural networks robust to label noise: A loss correction approach," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, 2017, pp. 2233–2241.

[47] X.-B. Xia, T.-L. Liu, N.-N. Wang, B. Han, C. Gong, G. Niu, and M. Sugiyama, "Are anchor points really indispensable in label-noise learning?" in *Advances in Neural Information Processing Systems 32*, Vancouver, Canada, 2019, pp. 6835–6846.

[48] J.-Q. Lv, M. Xu, L. Feng, G. Niu, X. Geng, and M. Sugiyama, "Progressive identification of true labels for partial-label learning," in *Proceedings of the 37th International Conference on Machine Learning*, Virtual Event, 2020, pp. 6500–6510.

[49] Z.-L. Zhang and M. R. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *Advances in Neural Information Processing Systems 31*, Montréal, Canada, 2018, pp. 8792–8802.

[50] Y. Katsura and M. Uchida, "Bridging ordinary-label learning and complementary-label learning," in *Proceedings of the 12th Asian Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, Bangkok, Thailand, 2020, pp. 161–176.

[51] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z.-M. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J.-J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, Vancouver, Canada, 2019, pp. 8024–8035.

[52] M.-L. Zhang and L. Wu, "Lift: Multi-label learning with label-specific features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 1, pp. 107–120, 2015.

[53] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, CA, 2015.

[54] L. Feng, J. Huang, S.-L. Shu, and B. An, "Regularized matrix factorization for multilabel learning with missing labels," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 3710–3721, 2022.

[55] C.-X. Wang, Y.-J. Lin, and J.-H. Liu, "Feature selection for multi-label learning with missing labels," *Appl. Intell.*, vol. 49, no. 8, pp. 3027–3042, 2019.

[56] M. D. Reid and R. C. Williamson, "Composite binary losses," *J. Mach. Learn. Res.*, vol. 11, pp. 2387–2422, 2010.

Yi Gao received the BSc and MSc degrees in computer science from Northwest University, China, in 2017 and 2020 respectively. She is currently pursuing the PhD degree in Southeast University, China. Her main research interests include machine learning and data mining, with a focus on learning from complementary labels.

Miao Xu is a senior lecturer in the School of Electrical Engineering and Computer Science at the University of Queensland, Australia. She was awarded the Australian Research Council Discovery Early Career Researcher Award (DECRA) in 2023. Dr Xu specializes in machine learning and data mining, particularly focusing on the challenges of learning from imperfect information. Dr Xu earned a PhD from Nanjing University, where research efforts led to notable recognitions including the CAAI Outstanding Doctoral Dissertation Award.

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 PAKDD'19, CCF-ICAI'19, ACML'17, CCFAI'17, PRICAI'16, Senior PC member or Area Chair of KDD 2021-2022, AAAI 2017-2020, IJCAI 2017-2023, ICDM 2015-2022, etc. He is also on the editorial board of IEEE Transactions on Pattern Analysis and Machine Intelligence, ACM Transactions on Intelligent Systems and Technology, Science China Information Sciences, Frontiers of Computer Science, Machine Intelligence Research, 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 AAAI, ACM, IEEE.

APPENDIX A
THE PROOF OF THEOREM 1

Theorem 1. *Given an instance $\boldsymbol{x}$, suppose Y is the relevant label set and the label l_j is the complementary label which is randomly selected. Then the following equality holds:*

$$p(\bar{y}^j = 1|\boldsymbol{x}) = \sum_{C\in\mathcal{Y}', l_j\notin C} p(\bar{y}^j = 1|Y = C)p(Y = C|\boldsymbol{x}) \geq \sum_{k=1,k\neq j}^{K} p(\bar{y}^j = 1|y^k = 1)p(y^k = 1|\boldsymbol{x}).$$

Proof. Firstly, we should introduce the addition rule of probability: $p(AB) = p(A) + p(B) - p(A \cup B)$, so we have $p(AB) \geq p(A) + p(B)$. We start to prove the above inequlity. According to the assumption: $p(\bar{y}|Y) = p(\bar{y}|Y, \boldsymbol{x})$, we have

$$\begin{aligned}
p(\bar{y}^j = 1|\boldsymbol{x}) &= \sum_{C\in\mathcal{Y}', l_j\notin C} p(\bar{y}^j = 1|Y = C)p(Y = C|\boldsymbol{x}) \\
&= \sum_{C\in\mathcal{Y}', l_j\notin C} p(\bar{y}^j = 1|Y = C, \boldsymbol{x})p(Y = C|\boldsymbol{x}) \\
&= \sum_{C\in\mathcal{Y}', l_j\notin C} p(\bar{y}^j = 1, Y = C|\boldsymbol{x}) \\
&= \sum_{C\in\mathcal{Y}', l_j\notin C} p(Y = C|\bar{y}^j = 1, \boldsymbol{x})p(\bar{y}^j = 1|\boldsymbol{x}).
\end{aligned}$$

According to addition rule of probability, so we have

$$\begin{aligned}
p(\bar{y}^j = 1|\boldsymbol{x}) &\geq \sum_{C\in\mathcal{Y}', l_j\notin C} \left[\sum_{k=1,k\neq j,l_k\in C}^{K} p(y^k = 1|\bar{y}^j = 1, \boldsymbol{x}) + \sum_{k=1,l_k\notin C}^{K} p(y^k = 0|\bar{y}^j = 1, \boldsymbol{x})\right] p(\bar{y}^j = 1|\boldsymbol{x}) \\
&\geq \sum_{C\in\mathcal{Y}', l_j\notin C} \sum_{k=1,k\neq j,l_k\in C}^{K} p(y^k = 1|\bar{y}^j = 1, \boldsymbol{x})p(\bar{y}^j = 1|\boldsymbol{x}) \qquad \because \sum_{k=1,l_k\notin C}^{K} p(y^k = 0|\bar{y}^j = 1, \boldsymbol{x}) \geq 0 \\
&= \sum_{C\in\mathcal{Y}', l_j\notin C} \sum_{k=1,k\neq j,l_k\in C}^{K} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) \\
&= \sum_{C\in\mathcal{Y}', l_j\notin C} \sum_{k=1,k\neq j}^{K} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) \qquad \because p(y^k = 1|\boldsymbol{x}) = 0 \text{ if } l_k \notin Y \\
&= \sum_{k=1,k\neq j}^{K} \sum_{C\in\mathcal{Y}', l_j\notin C} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) \\
&= \sum_{k=1,k\neq j}^{K} (2^{K-1} - 1)p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) \\
&\geq \sum_{k=1,k\neq j}^{K} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) \\
&= \sum_{k=1,k\neq j}^{K} p(\bar{y}^j = 1|y^k = 1)p(y^k = 1|\boldsymbol{x}).
\end{aligned}$$

□

APPENDIX B
THE PROOF OF THEOREM 3

Theorem 3. *Under a MLL scenario: suppose the labels $l_{z_1}, l_{z_2} \in \mathcal{Y}$ ($z_1, z_2 \in [K], z_1 \neq z_2$) are dependent, and the labels belonging to $\mathcal{Y} \setminus \{l_{z_1}, l_{z_2}\}$ are mutually exclusive. For any $\boldsymbol{x} \in \mathcal{X}$, its label set $Y \subseteq \{l_{z_1}, l_{z_2}\}$ and $Y \neq \emptyset$. Let the label l_j ($j \in [K], j \neq z_1, z_2$) be the complementary label of $\boldsymbol{x}$. $\mathbf{T}_{z_1 j}$ and $\mathbf{T}_{z_2 j}$ calculated from label correlations satisfy*

$$\begin{aligned}
\mathbf{T}_{z_1 j} &= \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(y^{z_1} = 1|\boldsymbol{x})}, \\
\mathbf{T}_{z_2 j} &= \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x})p(y^{z_2} = 1|\boldsymbol{x})},
\end{aligned}$$

where $[K]$ *denotes the integer set* $\{1, 2, \ldots, K\}$. *The difference of* $\mathbf{T}$ *and* $\mathbf{Q}$ *on the complementary label* l_j *is*

$$\ell_j \geq 2(\frac{1}{\xi^2} - 1)p(\bar{y}^j = 1|\boldsymbol{x}),$$

where $\xi = \max\{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x}), p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x})\}$.

Proof. We start calculating the difference ℓ_j from estimating the transition probabilities $\mathbf{T}_{z_1 j}$ and $\mathbf{T}_{z_2 j}$. According to Definition 2 and the description of Theorem 3, we have

$$\begin{aligned}
p(\bar{y}^j = 1|\boldsymbol{x}) = &\sum_{k=1,k\neq j,z_1,z_2}^{K} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) + p(\bar{y}^j = 1|y^{z_1} = 1, y^{z_2} = 1, \boldsymbol{x})p(y^{z_1} = 1, y^{z_2} = 1|\boldsymbol{x}) \\
&+ p(\bar{y}^j = 1|y^{z_1} = 1, y^{z_2} = 0, \boldsymbol{x})p(y^{z_1} = 1, y^{z_2} = 0|\boldsymbol{x}) + p(\bar{y}^j = 1|y^{z_1} = 0, y^{z_2} = 1, \boldsymbol{x})p(y^{z_1} = 0, y^{z_2} = 1|\boldsymbol{x}) \\
&+ p(\bar{y}^j = 1|y^{z_1} = 0, y^{z_2} = 0, \boldsymbol{x})p(y^{z_1} = 0, y^{z_2} = 0|\boldsymbol{x}) \\
= &\sum_{k=1,k\neq j,z_1,z_2}^{K} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) + p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 1, \boldsymbol{x})p(y^{z_1} = 1|\boldsymbol{x}) \\
&+ p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 0, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_2} = 0, \boldsymbol{x})p(y^{z_2} = 0|\boldsymbol{x}) \\
&+ p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 0, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 0, \boldsymbol{x})p(y^{z_1} = 0|\boldsymbol{x}) \\
&+ p(y^{z_2} = 0|\bar{y}^j = 1, y^{z_1} = 0, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 0, \boldsymbol{x})p(y^{z_1} = 0|\boldsymbol{x}).
\end{aligned}$$

Based on the assumption that $\bar{y}$ and $\boldsymbol{x}$ are conditionally independent given Y, then we can have

$$\begin{aligned}
p(\bar{y}^j = 1|\boldsymbol{x}) = &\sum_{k=1,k\neq j,z_1,z_2}^{K} p(\bar{y}^j = 1|y^k = 1)p(y^k = 1|\boldsymbol{x}) + p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 1)p(y^{z_1} = 1|\boldsymbol{x}) \\
&+ p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 0, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_2} = 0)p(y^{z_2} = 0|\boldsymbol{x}) \\
&+ p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 0, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 0)p(y^{z_1} = 0|\boldsymbol{x}) \\
&+ p(y^{z_2} = 0|\bar{y}^j = 1, y^{z_1} = 0, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 0)p(y^{z_1} = 0|\boldsymbol{x}).
\end{aligned}$$

Since $p(\bar{y}^j = 1|y^{z_1} = 0)$ and $p(\bar{y}^j = 1|y^{z_2} = 0)$ do not hold according to the definition of the transition matrix, we can obtain

$$\begin{aligned}
p(\bar{y}^j = 1|\boldsymbol{x}) = &\sum_{k=1,k\neq j,z_1,z_2}^{K} p(\bar{y}^j = 1|y^k = 1)p(y^k = 1|\boldsymbol{x}) + p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 1)p(y^{z_1} = 1|\boldsymbol{x}) \\
= &\, p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 1)p(y^{z_1} = 1|\boldsymbol{x}) \qquad \text{because } p(y^k = 1|\boldsymbol{x}) = 0 \text{ if } l_k \notin Y
\end{aligned}$$

$$\Rightarrow \mathbf{T}_{z_1 j} = p(\bar{y}^j = 1|y^{z_1} = 1) = \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(y^{z_1} = 1|\boldsymbol{x})}.$$

Similarly, we can get

$$\mathbf{T}_{z_2 j} = p(\bar{y}^j = 1|y^{z_2} = 1) = \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x})p(y^{z_2} = 1|\boldsymbol{x})}.$$

Next, we calculate the difference ℓ_j. The rest of the elements of $\mathbf{T}_{\cdot j}$ are same as those estimated by multi-class CLL. According the definition of ℓ_j, we have

$$\begin{aligned}
\ell_j &= \sum_{k=1}^{K} |\mathbf{T}_{kj} - \mathbf{Q}_{kj}| \\
&= \left|\mathbf{T}_{z_1 j} + \mathbf{T}_{z_2 j} - 2p(\bar{y}^j = 1|\boldsymbol{x})\right| \\
&= \left| \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(y^{z_1} = 1|\boldsymbol{x})} + \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x})p(y^{z_2} = 1|\boldsymbol{x})} - 2p(\bar{y}^j = 1|\boldsymbol{x}) \right| \\
&\geq \left| 2(\frac{1}{\xi^2} - 1)p(\bar{y}^j = 1|\boldsymbol{x}) \right| \\
&= 2(\frac{1}{\xi^2} - 1)p(\bar{y}^j = 1|\boldsymbol{x}). \quad \because \frac{1}{\xi^2} \geq 1
\end{aligned}$$

Because $0 \leq p(y^{z_1} = 1|\boldsymbol{x}) \leq p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x}) \leq 1$ and $0 \leq p(y^{z_2} = 1|\boldsymbol{x}) \leq p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x}) \leq 1$, ξ is defined as $\xi = \max\{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x}), p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x})\}$, the above inequation holds. □

APPENDIX C
THE PROOF OF COROLLARY 4

Corollary 4. *Under a MLL scenario: there are m ($m \geq 2$) labels $l_{z_1}, l_{z_2}, \ldots, l_{z_m} \in \mathcal{Y}$ ($z_1, \ldots, z_m \in [K]$) that are dependent, while the labels belong to $\mathcal{Y} \setminus \{l_{z_1}, l_{z_2}, \ldots, l_{z_m}\}$ are mutually exclusive. For any $\boldsymbol{x} \in \mathcal{X}$, its relevant set $Y \subseteq \{l_{z_1}, l_{z_2} \ldots, l_{z_m}\}$ and $Y \neq \emptyset$. Suppose the label l_j is the complementary label of $\boldsymbol{x}$. The difference ℓ_j between $\mathbf{T}$ and $\mathbf{Q}$ has*

$$\ell_j \geq m(\frac{1}{\xi^m} - 1)p(\bar{y}^j = 1|\boldsymbol{x}),$$

where $\xi = \max\{p(y^{z_m} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{m-1}} = 1, \boldsymbol{x}), p(y^{z_{m-1}} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{m-2}} = 1, y^{z_m} = 1, \boldsymbol{x}), \ldots, p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \ldots, y^{z_m} = 1, \boldsymbol{x})\}$ ($\xi \in (0, 1]$).

Proof. Here, we apply induction to compute the difference as m increases. We start by computing the difference in the case of $m = 3$. Suppose class labels $l_{z_1}, l_{z_2}, l_{z_3} \in \mathcal{Y}$ are dependent, while the rest of labels in the label space are mutually exclusive. $\boldsymbol{x}$ is associated with $Y \subseteq \{l_{z_1}, l_{z_2}, l_{z_3}\}$, and $Y \neq \emptyset$. Then we calculate transition probabilities in $\mathbf{T}$ from label correlations according to Theorem 3 as follows:

$$\begin{aligned}
p(\bar{y}^j = 1|\boldsymbol{x}) &= \sum_{k=1, k\neq j, z_1, z_2, z_3}^{K} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) + p(\bar{y}^j = 1, y^{z_1} = 1, y^{z_2} = 1, y^{z_3} = 1|\boldsymbol{x}) \\
&= p(\bar{y}^j = 1, y^{z_1} = 1, y^{z_2} = 1, y^{z_3} = 1|\boldsymbol{x}) \\
&= p(y^{z_3} = 1|\bar{y}^j = 1, y^{z_1} = 1, y^{z_2} = 1, \boldsymbol{x})p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 1, \boldsymbol{x})p(y^{z_1} = 1|\boldsymbol{x}) \\
&= p(y^{z_3} = 1|\bar{y}^j = 1, y^{z_1} = 1, y^{z_2} = 1, \boldsymbol{x})p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 1)p(y^{z_1} = 1|\boldsymbol{x})
\end{aligned}$$

$$\Rightarrow \mathbf{T}_{z_1 j} = p(\bar{y}^j = 1|y^{z_1} = 1) = \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_3} = 1|\bar{y}^j = 1, y^{z_1} = 1, y^{z_2} = 1, \boldsymbol{x})p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(y^{z_1} = 1|\boldsymbol{x})}.$$

$\mathbf{T}_{z_2 j}$ and $\mathbf{T}_{z_3 j}$ use the same way to estimate. Due to $0 \leq p(y^{z_1} = 1|\boldsymbol{x}) \leq p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \boldsymbol{x}) \leq p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, y^{z_3} = 1, \boldsymbol{x}) \leq 1$, let $\xi = \max\{p(y^{z_3} = 1|\bar{y}^j = 1, y^{z_1} = 1, y^{z_2} = 1, \boldsymbol{x}), p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, y^{z_3} = 1, \boldsymbol{x}), p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, y^{z_3} = 1, \boldsymbol{x})\}$, we can obtain

$$\mathbf{T}_{z_1 j} = p(\bar{y}^j = 1|y^{z_1} = 1) \geq \frac{1}{\xi^3} p(\bar{y}^j = 1|\boldsymbol{x}).$$

Similarly, we can compute $\mathbf{T}_{z_2 j}, \mathbf{T}_{z_3 j} \geq \frac{1}{\xi^3} p(\bar{y}^j = 1|\boldsymbol{x})$. Then the difference ℓ_j is

$$\begin{aligned}
\ell_j &= \sum_{k=1}^{K} |\mathbf{T}_{kj} - \mathbf{Q}_{kj}| \\
&= \left|\mathbf{T}_{z_1 j} + \mathbf{T}_{z_2 j} + \mathbf{T}_{z_3 j} - 3p(\bar{y}^j = 1|\boldsymbol{x})\right| \\
&\geq 3(\frac{1}{\xi^3} - 1)p(\bar{y}^j = 1|\boldsymbol{x}).
\end{aligned}$$

Similarly, for any m ($0 < m < K$), suppose class labels $l_{z_1}, l_{z_2}, \ldots, l_{z_m} \in \mathcal{Y}$ are strongly dependent, while the rest of labels in the label space are mutually exclusive. $\boldsymbol{x}$ is associated with $Y \subseteq \{l_{z_1}, l_{z_2}, l_{z_3}\}$ and $Y \neq \emptyset$. Then we calculate transition probabilities from label correlations:

$$\begin{aligned}
p(\bar{y}^j = 1|\boldsymbol{x}) &= \sum_{k=1, k\neq j, z_1, \ldots, z_m}^{K} p(\bar{y}^j = 1|y^k = 1, \boldsymbol{x})p(y^k = 1|\boldsymbol{x}) + p(\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_m} = 1|\boldsymbol{x}) \\
&= p(\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_s} = 1|\boldsymbol{x}) \\
&= p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(\bar{y}^j = 1|y^{z_1} = 1)p(y^{z_1} = 1|\boldsymbol{x})\Pi_{i=3}^{m} p(y^{z_i} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{i-1}} = 1, \boldsymbol{x})
\end{aligned}$$

$$\Rightarrow \mathbf{T}_{z_1 j} = p(\bar{y}^j = 1|y^{z_1} = 1) = \frac{p(\bar{y}^j = 1|\boldsymbol{x})}{p(y^{z_2} = 1|\bar{y}^j = 1, y^{z_1} = 1, \boldsymbol{x})p(y^{z_1} = 1|\boldsymbol{x})\Pi_{i=3}^{m} p(y^{z_i} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{i-1}} = 1, \boldsymbol{x})}.$$

As discussed above, $\mathbf{T}_{z_1 j} \geq \frac{1}{\xi^m} p(\bar{y}^j = 1|\boldsymbol{x})$ since $\xi = \max\{p(y^{z_m} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{m-1}} = 1, \boldsymbol{x}), p(y^{z_{m-1}} = 1|\bar{y}^j = 1, y^{z_1} = 1, \ldots, y^{z_{m-2}} = 1, y^{z_m} = 1, \boldsymbol{x}), \ldots, p(y^{z_1} = 1|\bar{y}^j = 1, y^{z_2} = 1, \ldots, y^{z_m} = 1, \boldsymbol{x})\}$ ($\xi \in (0, 1]$). Using the same calculation way, we can obtain $\mathbf{T}_{z_2 j}, \ldots, \mathbf{T}_{z_m j} \geq \frac{1}{\xi^m} p(\bar{y}^j = 1|\boldsymbol{x})$. Based on induction, we can summarize the difference $\ell_j = \sum_{k=1}^{K} |\mathbf{T}_{kj} - \mathbf{Q}_{kj}| \geq m(\frac{1}{\xi^m} - 1)p(\bar{y}^j = 1|\boldsymbol{x})$. □

APPENDIX D
THE PROOF OF THEOREM 6

Theorem 6. *With Assumption 5, suppose the transition matrix* $\mathbf{T}$ *is invertible, then the ML-CLL optimal classifier* $\boldsymbol{f}^*_{CL}$ *converges to the MLL optimal classifier* $\boldsymbol{f}^*$.

Proof. Before presenting the proof, we need to introduce the definition of *proper composite losses* as defined in [56]. According to the definition, many losses are considered *composite*, comprising a loss and a *link function* denoted as ψ (where ψ is invertible). Let L_ψ be a composite loss, which can be expressed with the assistance of a link function as follows:

$$L_\psi(\boldsymbol{g}(\boldsymbol{x}), \boldsymbol{y}) = L(\psi^{-1}(\boldsymbol{g}(\boldsymbol{x})), \boldsymbol{y}),$$

where the *inverse* link function (ψ^{-1}) represents the *sigmoid* function in the case of *binary cross-entropy* (BCE) loss. In this context, $\boldsymbol{g}(\cdot) \in \mathbb{R}^K$ denotes the output of a neural network before the application of the *sigmoid* function, and $g^k(\cdot)$ denotes the k-th element of this output. Within neural networks, the real-valued function $\boldsymbol{f}(\boldsymbol{x})$ denotes the output of $\boldsymbol{g}(\boldsymbol{x})$ after passing through the *sigmoid* function. Moreover, [56] introduced the property of composite losses when the composite loss is considered *proper*, which means:

$$\operatorname*{argmin}_{\boldsymbol{g}} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}[L_\psi(\boldsymbol{g}(\boldsymbol{x}), \boldsymbol{y})] = \psi(p(\boldsymbol{y}|\boldsymbol{x})).$$

In cases where composite losses are deemed *proper*, their minimizer exhibits a specific form corresponding to the link function applied to the class-conditional probabilities $p(\boldsymbol{y}|\boldsymbol{x})$. It's noteworthy that *binary cross-entropy* (BCE) loss and square loss are both examples of *proper composite* losses [46].
With Eq. (8) and the preceding equations, it is straightforward to have

$$\begin{aligned}\bar{L}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) &= L(\mathbf{T}^T\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}}) \\ &= L(\mathbf{T}^T\psi^{-1}(\boldsymbol{g}(\boldsymbol{x})), \bar{\boldsymbol{y}}) \\ &= L_\phi(\boldsymbol{g}(\boldsymbol{x}), \bar{\boldsymbol{y}}),\end{aligned}$$

where ϕ^{-1} refers to the inverse link function, and we denote $\phi^{-1} = \psi^{-1} \circ \mathbf{T}^T$. Since ϕ^{-1} is composited by invertible functions, ϕ^{-1} is also invertible. Equivalently, $\phi = (\mathbf{T}^{-1})^T \circ \psi$. Leveraging the property of proper composite losses, the minimizer of the expected risk is:

$$\begin{aligned}\operatorname*{argmin}_{\boldsymbol{g}} \mathbb{E}_{(\boldsymbol{x},\bar{\boldsymbol{y}})\sim\bar{\mathcal{D}}}[L_\phi(\boldsymbol{g}(\boldsymbol{x}), \bar{\boldsymbol{y}})] &= \phi(p(\bar{\boldsymbol{y}}|\boldsymbol{x})) \\ &= \psi((\mathbf{T}^{-1})^T p(\bar{\boldsymbol{y}}|\boldsymbol{x})) \\ &= \psi((\mathbf{T}^{-1})^T\mathbf{T}^T p(\boldsymbol{y}|\boldsymbol{x})) \\ &= \psi(p(\boldsymbol{y}|\boldsymbol{x})) \\ &= \operatorname*{argmin}_{\boldsymbol{g}} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}[L_\psi(\boldsymbol{g}(\boldsymbol{x}), \boldsymbol{y})].\end{aligned}$$

Since the *sigmoid* function preserves the rank of its inputs, and $\boldsymbol{f}(\boldsymbol{x})$ is the output of $\boldsymbol{g}(\boldsymbol{x})$ after passing through the *sigmoid* function, it holds for that

$$\begin{aligned}\operatorname*{argmin}_{\boldsymbol{f}} \mathbb{E}_{(\boldsymbol{x},\bar{\boldsymbol{y}})\sim\bar{\mathcal{D}}}[\bar{L}(\boldsymbol{f}(\boldsymbol{x}), \bar{\boldsymbol{y}})] &= \operatorname*{argmin}_{\boldsymbol{f}} \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}}[L(\boldsymbol{f}(\boldsymbol{x}), \boldsymbol{y}] \\ \Leftrightarrow \operatorname*{argmin}_{\boldsymbol{f}} R_{\bar{L}}(\boldsymbol{f}) &= \operatorname*{argmin}_{\boldsymbol{f}} R_L(\boldsymbol{f}).\end{aligned}$$

Finally, it can be observed that $\boldsymbol{f}^*_{CL}$ converges to $\boldsymbol{f}^*$, where $\boldsymbol{f}^*_{CL}$ and $\boldsymbol{f}^*$ are minimizers of $R_{\bar{L}}(\boldsymbol{f})$ and $R_L(\boldsymbol{f})$, respectively. □