MulCo: Representation Learning for Multiple Complementary Labels

YI GAO and YUAN-YUAN MENG, School of Computer Science and Engineering, and Key Laboratory of Computer Network and Information Integration, Southeast University, Nanjing, China
MIAO XU, The University of Queensland, Brisbane, Australia
MIN-LING ZHANG, School of Computer Science and Engineering, and Key Laboratory of Computer Network and Information Integration, Southeast University, Nanjing, China

Multiple complementary-label learning (MCLL) is a machine learning task that involves learning a classifier from instances with *multiple complementary labels* (MCLs). MCLs are labels that indicate the incorrect labels of an instance. Previous methods for learning with ambiguous supervised information may not be effective because MCLs only make up a small proportion of all labels. In this article, we propose MulCo, a simple yet effective framework that uses contrastive learning to enhance the representation capability in MCLL. Contrastive learning involves contrasting semantically similar and dissimilar pairs of instances, with the goal of benefiting from negatives whose ground-truth labels differ from those of anchors. However, it is possible for dissimilar pairs to have the same label due to the random sampling of negatives from inaccurately labeled data. To solve this problem, we design a sifted contrastive loss for MulCo to correct the sampling of same-label negative pairs. We also provide theoretical evidence for the feasibility of the sifted contrastive loss by establishing an upper bound on the ideal contrastive loss. Correspondingly, we develop two progressive solutions using the properties of complementary labels to approximate the ideal contrastive loss through weighting. Our empirical study demonstrates the effectiveness of the proposed method. The code of this article is available at https://github.com/gaoyi439/MulCo.

CCS Concepts: • **Computing methodologies** → **Learning paradigms**;

Additional Key Words and Phrases: Weakly supervised learning, Complementary label learning, Contrastive learning

Associate Editor: Carl Yang

ACM Reference format:
Yi Gao, Yuan-Yuan Meng, Miao Xu, and Min-Ling Zhang. 2026. MulCo: Representation Learning for Multiple Complementary Labels. *ACM Trans. Knowl. Discov. Data.* 20, 5, Article 80 (May 2026), 25 pages.
https://doi.org/10.1145/3806654

This work was supported by the National Natural Science Foundation of China (62225602, 624B2042) and Basic Research Program of Jiangsu (BK20253021). Miao Xu is supported by the Australian Research Council (DE230101116).
Authors' Contact Information: Yi Gao, School of Computer Science and Engineering, and Key Laboratory of Computer Network and Information Integration, Southeast University, Nanjing, China; e-mail: gao_yi@seu.edu.cn; Yuan-Yuan Meng, School of Computer Science and Engineering, and Key Laboratory of Computer Network and Information Integration, Southeast University, Nanjing, China; e-mail: mengyy@seu.edu.cn; Miao Xu, The University of Queensland, Brisbane, Australia; e-mail: miao.xu@uq.edu.au; Min-Ling Zhang (corresponding author), School of Computer Science and Engineering, and Key Laboratory of Computer Network and Information Integration, Southeast University, Nanjing, China; e-mail: zhangml@seu.edu.cn.

ACM 1556-472X/2026/5-ART80
https://doi.org/10.1145/3806654

1 Introduction

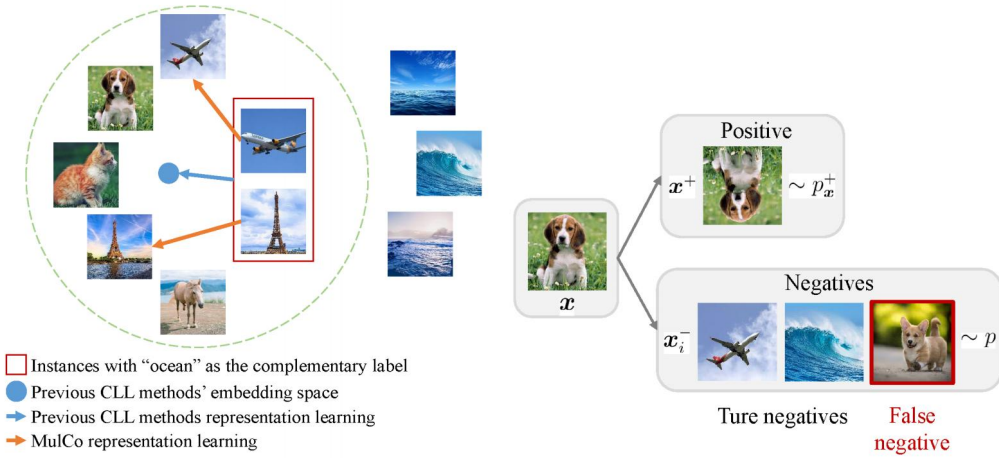
Multi-class classification tasks [17] typically require a large amount of high-quality labeled data, which can be difficult and costly to obtain [8, 31], especially for private data. The development of a weakly supervised learning framework called **Complementary Label Learning (CLL)** has helped to alleviate this issue by allowing instances to be associated with complementary labels, which specify the incorrect labels of the instance [7, 10, 15, 16, 35]. Collecting complementary labels is generally easier than collecting ordinary labels, such as when asking private questions [9, 15, 16]. CLL methods can be divided into two categories based on the number of complementary labels per instance: (1) learning with a single complementary label [4, 10, 15, 45, 46] and (2) learning with **Multiple Complementary Labels (MCLs)** [7, 19]. The latter, known as **Multiple Complementary-Label Learning (MCLL)**, is the focus of this article.

Complementary labels are labels that are assigned to an instance in addition to its ground-truth label [4, 11]. They provide negative feedback by indicating which labels an instance should not be classified as and are used to help the model learn more accurate representations during training [4, 7, 10, 15, 21]. However, the use of complementary labels can lead to label ambiguity, where it is unclear which label an instance should be classified as, due to the small number of complementary labels relative to the label space. This can result in imprecise representations, as existing methods only push instances away from the complementary labels, but do not provide guidance on which class an instance should belong to [4, 7, 15, 16, 46]. As a result, training may be led by ambiguous learning directions, and instances are mapped to similar embedding spaces, even if their ground-truth labels are different (as shown in Figure 1(a)).

Contrastive loss has gained popularity as a powerful tool for representation learning [3, 5, 13, 24, 44] because it can distinguish between semantically similar and dissimilar pairs of instances [5, 28]. This leads to representations of similar instances being close together and those of dissimilar instances being far apart in the embedding space, resulting in more precise representations. In unsupervised contrastive learning, data points serve as anchors, and positive pairs are composed of one data point and its augmentation, while negative pairs are composed of one data point and the augmentation of another point. However, this method of sampling pairs can lead to negative pairs being composed of data points from the same class, resulting in false negatives that degrade performance in subsequent classification tasks [5, 20, 28]. Additionally, true negative pairs can guide a learning method to correct mistakes more quickly, leading to faster convergence to precise representations [5, 28, 30]. To address the issue of sampling bias, labels are provided in the training data to ensure that true negative pairs are generated. However, when this principle is applied directly to MCLL, the negatives of an anchor may be uniformly drawn from inaccurately labeled data, leading to negatives having the same label as the anchor, as illustrated in Figure 1(b) and resulting in imprecise representations [5, 28, 30].

To address the issue of sampling bias in MCLL, we propose *MulCo* (*Mul*tiple CLL with *Co*ntrastive representation disambiguation) to obtain high-quality representations in MCLL, as demonstrated in Figure 1(a). MulCo uses a sifted contrastive loss that uses predictive labels to directly sample true negatives (negatives with different ground-truth labels than the anchor) from the training data in MCLL. By using true negatives to guide the learning direction, MulCo is able to quickly converge to a solution with distinguishable representations, leading to improved classification performance. Theoretically, it has been shown that this contrastive loss is an upper bound on the ideal loss, whose negatives are sampled from data with truly different labels than the anchor.

The method described above for sampling true negatives may introduce ambiguities during optimization if the classifier's predictions are not accurate. To address this issue, we propose two solutions that gently approximate the ideal contrastive loss. These solutions correct the sampling



(a) Previous CLL methods vs. MulCo (b) The definition of *"sampling bias"* [5]

Fig. 1. For (a), two instances with "plane" and "architecture," respectively, in the red box, both select the label "ocean" as their complementary label. Previous CLL methods only can push these two instances far away from instances with the label "ocean" during the representation learning process, but they fail to know which of these two instances should be close to. This could result in these two instances being with different labels but still mapped in a similar embedding space. Instead, MulCo, respectively pulls together these two instances and their similar instances in the embedding space. This is because MulCo depends on contrastive learning, which has a more explicit representation learning direction than previous CLL methods. For (b), "*sampling bias*" refers to: negatives $\boldsymbol{x}_i^-$ are uniformly drawn from the distribution of training data p, which may lead to $\boldsymbol{x}_i^-$ are actually similar to the anchor $\boldsymbol{x}$, such as false negative.

bias by dynamically weighting the complementary labels based on their properties. In summary, the main contributions of this article are as follows:

- —We explore contrastive learning for MCLL and propose a simple yet effective framework called MulCo to distinguish learning direction to facilitate more precise representations.
- —To alleviate sampling bias, we propose a sifted contrastive loss for MulCo based on predictive labels to sample true negatives and theoretically prove it is an upper bound on the ideal contrastive loss. In addition, two progressive solutions are proposed to correct the sampling bias, which can prevent filtration error caused by inexact predictions in the sifted contrastive loss.
- —Empirical study shows that our proposed framework MulCo obtains comparable or competitive performance with **State-of-the-Art (SOTA)** baselines.

The rest of this article is organized as follows. Section 2 is used to briefly review the related work of this article. Section 3 gives a background of MCLL. Section 4 introduces the proposed framework with theoretical analyses and algorithmic details. Experiments and conclusion are stated in Sections 5 and 6, respectively.

2 Related Work

In this section, we briefly review related work on CLL and contrastive learning.

2.1 CLL

CLL is a weakly supervised learning scenario proposed in recent years, which aims to learn a classifier that can predict ground-truth labels for unseen instances from less expensive annotated

data [15, 19, 49]. According to the number of complementary labels for each instance, previous CLL methods can be roughly grouped into two categories: (1) learning with a single complementary label [4, 10, 15, 16, 45, 46] and (2) learning with MCLs [1, 7, 35].

CLL is first proposed by Ishida et al. [15] to solve the problem of learning with a single complementary label, which uses the uniform generation assumption to rewrite one-versus-all and pairwise comparison loss functions [47]. Besides, the binary loss functions $\ell(z) : \mathbb{R} \to \mathbb{R}^+$ used in OVA and PC loss need to satisfy the symmetric condition: $\ell(z) + \ell(-z) = 1$. To break free from the limits of loss functions, Ishida et al. [16] propose a general framework that is available to arbitrary models and loss functions. Subsequently, methods applying a transition matrix are developed to solve a **Single CLL (SCLL)** problem, which recover ground-truth labels from complementary labels by the estimated transition matrix [45, 46]. For previous work [4, 10], the property that expects the predictive probabilities of complementary labels to zero is used to design reasonable loss functions for solving the SCLL problem.

The second category provides more supervised information than the first category, where per instance is associated with MCLs. MCLL is initially proposed by Feng et al. [7], who solve MCLL problem by supposing the generation relationship between ground-truth labels and MCLs. And they design upper-bounded losses for MCLL to prevent overfitting issues. Wang et al. [35] use consistency regularization with data augmentation to improve MCLL performance. In addition, Cao et al. [1] combine MCLL and unlabeled learning to explore new learning scenarios. Nonetheless, few efforts have been made to improve representations based on distinguishing learning direction in CLL.

2.2 Contrastive Learning

Contrastive learning achieves prominent performance on representation learning through using instance similarity/dissimilarity [29, 39, 40, 43]. Most works devote to study sampling strategies for generating positive pairs and push forward this field significantly [3, 13, 24, 32, 33]. For example, Chen et al. [3] extensively study the impact of various data augmentation methods on contrastive learning performance.

Negative instances are randomly sampled from training data, which inevitably yields negative pairs that have similar semantic meanings. In fact, these negative pairs with similar semantic meanings should be closer in the embedding space, but they are pushed apart in the standard contrastive loss. To solve this problem, Li et al. [24] propose prototypical contrastive learning combined with clustering to capture the semantic structures of data and encode that into the learned embedding space. On the other hand, Khosla et al. [20] use supervised information to design supervised contrastive learning, which clusters data from the same class as the positive set. Inspired by supervised contrastive learning, contrastive learning is applied for many weakly supervised learning tasks, including noisy label learning [2, 41], semi-supervised learning [23, 38], and **Partial Label Learning (PLL)** [37]. Though some works point out that negative pairs may share similar semantic meanings, they solve this problem by constructing positive pairs. Actually, part of the essential cause for this problem samples negative instances from the training data distribution, while most of the current work ignores that.

3 Background

We formalize the problem setting of MCLL as follows. Let $\mathcal{X}$ be the input space and $\mathcal{Y} = \{1, 2, \ldots, K\}$ be the label space with K ($K > 2$) classes. An MCLL instance $\boldsymbol{x}$ is associated with a set of complementary labels $\bar{Y}$. $(\boldsymbol{x}, \bar{Y})$ is independently sampled from an unknown joint distribution $p(\boldsymbol{x}, \bar{Y})$. For convenience, $\bar{Y}$ can be represented by the vector $\bar{\boldsymbol{y}} \in \{0, 1\}^K$, where the ith element $\bar{y}^i$ equals 1 when $i \in \bar{Y}$ and 0 otherwise. It is obvious that MCLL will degrade into a SCLL learning when $\bar{Y}$ just contains one complementary label. In addition, if $\bar{Y}$ contains $K - 1$ complementary labels,

MCLL becomes an ordinary multi-class classification task. We exclude the special cases of $\bar{Y} = \emptyset$ or $\mathcal{Y}$ to ensure the MCLL problem hold, hence $\bar{Y} \in \bar{\mathcal{Y}}$ where $\bar{\mathcal{Y}} = \{2^{\mathcal{Y}} - \emptyset - \mathcal{Y}\}$ and $|\bar{\mathcal{Y}}| = 2^K - 2$.

The goal of MCLL is to learn a classifier $\boldsymbol{f} : \mathcal{X} \to \mathbb{R}^K$ which estimates the probability vector $p(\boldsymbol{y}|\boldsymbol{x})$, where $\boldsymbol{y}$ is a one-hot vector to denote the ground-truth label y of $\boldsymbol{x}$. Similar to the ordinary multi-class classification, MCLL expects to derive a function to assign labels for unseen instances. Let $h : \mathcal{X} \to \mathcal{Y}$ be the function assigning labels, which typically is defined as

$$h(\boldsymbol{x}) = \arg\max_{j \in \mathcal{Y}} f_j(\boldsymbol{x}). \tag{1}$$

where $f_j(\boldsymbol{x})$ refers to the jth element of $\boldsymbol{f}(\boldsymbol{x})$. It estimates the probability of the jth label being the ground-truth label given $\boldsymbol{x}$. MCLL is first proposed by Feng et al. [7], in which they designed a loss function motivated by maximizing predictive probabilities for all labels other than complementary ones, since complementary labels are incorrect labels for corresponding instances and provide negative feedback. The loss is expressed as

$$\bar{L}_{\log} = \mathbb{E}_{(\boldsymbol{x},\bar{Y})\sim p(\boldsymbol{x},\bar{Y})} \left[-\log \sum_{j=1, j\notin\bar{Y}}^{K} f_j(\boldsymbol{x}) \right]. \tag{2}$$

Due to $\sum_{j=1}^{K} f_j(\boldsymbol{x}) = 1$, Equation (2) can be rewritten as

$$\bar{L}_{\log} = \mathbb{E}_{(\boldsymbol{x},\bar{Y})\sim p(\boldsymbol{x},\bar{Y})} \left[-\log \left(1 - \sum_{j=1, j\in\bar{Y}}^{K} f_j(\boldsymbol{x}) \right) \right]. \tag{3}$$

From Equation (3), it is clear that minimizing $\bar{L}_{\log}$ is essentially minimizing predictive probabilities of complementary labels of $\boldsymbol{x}$. Although methods minimizing predictive probabilities of complementary labels are SOTA techniques to solve CLL problems, they may suffer from the ambiguous supervised information provided, which could not need precisely learned representation. In the following, we focus on designing a new MCLL learning framework named MulCo according to the principle of contrastive learning, which will learn representations from data in a more precise and effective way.

4 Method

In this section, we first introduce the proposed framework MulCo and the sifted contrastive loss. Then, we theoretically analyze the feasibility of the proposed sifted contrastive loss and present two progressive solutions that can more mildly approximate the ideal contrastive loss.

4.1 Representation Learning for MCLL

Label ambiguity posits an obstacle for MCLL to provide precise representations because ambiguous supervised information fails to derive an explicit learning direction. To solve this problem, in MulCo, we combine the loss of Equation (2) and a contrastive term that promotes the effectiveness of representations. While contrastive learning has been widely studied recently, it remains unexplored in MCLL.

Suppose $(\boldsymbol{x}, \boldsymbol{x}^+)$ is semantically similar (positive) pairs of instances, where an anchor $\boldsymbol{x}$ is drawn from a data distribution $p(\boldsymbol{x})$ (p for short) over $\mathcal{X}$. Positive pairs $(\boldsymbol{x}, \boldsymbol{x}^+)$ have the same label. $p^+_{\boldsymbol{x}}(\boldsymbol{x}') = p(\boldsymbol{x}'|h(\boldsymbol{x}') = h(\boldsymbol{x}))$ ($p^+_{\boldsymbol{x}}$ for short) denotes the distribution over observing an instance $\boldsymbol{x}'$ with the same label of $\boldsymbol{x}$. Similarly, the distribution over points with different labels of $\boldsymbol{x}$ is $p^-_{\boldsymbol{x}}(\boldsymbol{x}') = p(\boldsymbol{x}'|h(\boldsymbol{x}') \neq h(\boldsymbol{x}))$ ($p^-_{\boldsymbol{x}}$ for short). The goal is to learn a feature embedding function $\boldsymbol{g} : \mathcal{X} \to \mathbb{R}^d$ that maps $\boldsymbol{x}$ to a hypersphere with radius $1/\tau$, where τ is a temperature scaling

hyperparameter (WLOG, we set $\tau = 1$ for all theoretical analysis). Conventional contrastive learning achieves this goal by optimizing the standard contrastive loss [3, 13], which is defined as:

$$L_{\text{std}} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p,\boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\boldsymbol{x}_i^-\sim p} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \sum_{i=1}^N e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}} \right], \tag{4}$$

where the embeddings $\boldsymbol{q} = \boldsymbol{g}(\boldsymbol{x})$, $\boldsymbol{k}^+ = g'(\boldsymbol{x}^+)$ and $\boldsymbol{k}_i^- = \boldsymbol{g}'(\boldsymbol{x}_i^-)$, in which $\{\boldsymbol{x}_i^-\}_{i=1}^N$ refers to N negatives of $\boldsymbol{x}$ and $\boldsymbol{g}'(\cdot)$ denotes a momentum encoder that is updated by the parameter of $\boldsymbol{g}(\cdot)$ with a momentum. $\boldsymbol{x}_i^-$ is uniformly sampled from p, which means it is possible that $\boldsymbol{x}_i^-$ has the same label of $\boldsymbol{x}$. This naturally causes the conventional contrastive learning to suffer a performance drop from sampling bias [5], because it benefits from true negative pairs by pushing apart negative pairs $(\boldsymbol{x}, \boldsymbol{x}_i^-)$. To alleviate sampling bias in the standard contrastive loss, we propose a sifted way to filtrate true negatives based on predictive labels of instances given by the function h. Our sifted contrastive loss is expressed as:

$$\bar{L}_{\text{sifted}} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p,\boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\boldsymbol{x}_i^-\sim p} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \sum_{i=1}^N \mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}_i^-))} e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}} \right], \tag{5}$$

where $\mathbb{I}_{(\cdot)}$ be the indicator function, $\mathbb{I}_{(\cdot)} = 1$ when $\cdot$ is true otherwise 0. $h(\boldsymbol{x})$ and $h(\boldsymbol{x}_i^-)$ denote predictive labels of anchor instance $\boldsymbol{x}$ and negative instance $\boldsymbol{x}_i^-$ respectively. In $\bar{L}_{\text{sifted}}$, we define true negatives of $\boldsymbol{x}$ to be instances with different prediction labels, i.e., $h(\boldsymbol{x}) \neq h(\boldsymbol{x}_i^-)$. Despite its simplicity, we theoretically analyze the feasibility of this choosing strategy in Section 4.2, which drives a good empirical performance (Section 5). Finally, the optimization objective of MulCo is combining typical MCLL loss and the proposed sifted contrastive loss, which is expressed as:

$$\bar{\mathcal{L}}_{\text{sifted}} = \bar{L}_{\log} + \lambda \bar{L}_{\text{sifted}}, \tag{6}$$

where λ is a tradeoff parameter, we will introduce its selection method in Section 5. In this manner, MulCo can effectively reduce the negative impact of sampling bias to provide more precise representations according to the distinguishing learning direction.

Note that our main network architecture follows the principle of a popular contrastive learning method—MoCo [13] excepted the contrastive loss term. Remaining MCLL without ground-truth labels in practice, our method and other contrastive losses only are drawn from the data distribution and a surrogate positive distribution generated by data augmentations [3, 5, 13, 37].

4.2 Theoretical Analysis

As discussed above, sampling bias can result in a performance drop of contrastive learning empirically [5], so the sifted contrastive loss (Equation (5)) is designed to overcome this problem. In this subsection, we intend to theoretically analyze the feasibility of $\bar{L}_{\text{sifted}}$. Before the analysis, we need to state the ideal contrastive loss that is no trouble with sampling bias, because its positive and negative pairs belong to the desired labels. Intuitively, we can explain the feasibility of $\bar{L}_{\text{sifted}}$ by exploring the relationship between $\bar{L}_{\text{sifted}}$ and the ideal objective. Here, we utilize the definition of the ideal contrastive loss by [5, 28] and express it as follows:

$$L_{\text{ideal}} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p,\boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\boldsymbol{x}_i^-\sim p_{\boldsymbol{x}}^-} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{Q}{N}\sum_{i=1}^N e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}} \right], \tag{7}$$

where Q is a weighting parameter. Differing from the standard contrastive loss L_{std}, negatives of the ideal contrastive loss are sampled from $p_{\boldsymbol{x}}^-$. Nonetheless, $p_{\boldsymbol{x}}^-$ is unavailable since ground-truth labels are unknown in MCLL. Therefore, we design $\bar{L}_{\text{sifted}}$ to approximate the ideal contrastive loss L_{ideal} and derive $\bar{L}_{\text{sifted}}$ as an upper bound on L_{ideal} with Proposition 1.

Proposition 1. *If a label y is randomly selected from a uniform distribution over the label space $\mathcal{Y}$ with K possible labels, and each instance $\boldsymbol{x}$ has exactly one ground-truth label. Then the probability of y being the ground-truth label for a given instance $\boldsymbol{x}$ is $1/K$.*

According to Proposition 1, the data distribution can be decomposed into a linear combination of two probability density functions:

$$p(\boldsymbol{x}') = \frac{1}{K} p_{\boldsymbol{x}}^{+}(\boldsymbol{x}') + \frac{K-1}{K} p_{\boldsymbol{x}}^{-}(\boldsymbol{x}'). \tag{8}$$

With Proposition 1 and Equation (8), Theorem 2 shows that $\bar{L}_{\text{sifted}}$ is the upper bound of L_{ideal}.

Theorem 2. *For $\boldsymbol{f}$, there exists a parameter assignment θ to make $\boldsymbol{f}$ a perfect classifier, such that for any $\boldsymbol{x}$, $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$. Then h can assign the ground-truth label for $\boldsymbol{x}$. For any embedding function $\boldsymbol{g}$, fixed Q and $N \to \infty$, it holds that*

$$\bar{L}_{\text{sifted}} \geq L_{\text{ideal}}.$$

The proof is provided in Appendix A. Although the $\boldsymbol{x}^-$ in $\bar{L}_{\text{sifted}}$ is still drawn from p, we analyze its feasibility by establishing it is an upper bound on L_{ideal}. In addition, $\bar{L}_{\text{sifted}}$ will be close to L_{ideal} if h is a perfect function that can assign the label to unseen instances accurately. We theoretically prove that the sifted contrastive loss $\bar{L}_{\text{sifted}}$ is feasible and can effectively alleviate sampling bias, because minimizing an upper bound of the ideal objective is a reasonable and effective solution [5]. Recalling the lemma of [5] that states the standard contrastive loss L_{std} also being an upper bound on L_{ideal}, so why is our sifted contrastive loss $\bar{L}_{\text{sifted}}$ better than L_{std}? We will illustrate that in the following theorem.

Theorem 3. *There exists a parameter assignment θ which makes $\boldsymbol{f}$ for any $\boldsymbol{x}$ satisfy $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$, rendering it a perfect classifier. Then h can assign the ground-truth label for $\boldsymbol{x}$. For any embedding function $\boldsymbol{g}$ and $N \to \infty$, it holds that*

$$\bar{L}_{\text{sifted}} < L_{\text{std}} - \log\left(1 + \frac{e^{-2}}{K - 1 + K/N}\right).$$

The proof is stated in Appendix B. Recent works usually adopt large N (e.g., $N = 65,536$ in [13]), and the number of classes K is not a small integer, thus enabling the last term to be negligible. Theorem 3 shows that $\bar{L}_{\text{sifted}}$ is a lower bound on L_{std}. According to Theorem 2 and 3, our $\bar{L}_{\text{sifted}}$ is a tighter upper bound on the ideal contrastive loss L_{ideal} than L_{std}, which naturally derives that applying $\bar{L}_{\text{sifted}}$ will lead to better performance than L_{std} in MCLL. We also empirically demonstrate this in Section 5.

4.3 Progressive Solution

In the previous sections, we show that the sifted contrastive loss is effective in reducing negative impacts from sampling bias, which facilitates MulCo to provide precise representations with a distinguishing learning direction. Though the sifted contrastive loss adopts a straightforward strategy—choosing true negatives by label predictions—to alleviate the sampling bias problem, it is possible to produce new ambiguities during optimization if the classifier predicts inexactly. Therefore, we design two progressive solutions to avoid the rough way of the sifted contrastive loss, which corrects sampling bias via dynamic weighting based on the properties of complementary labels. These two progressive solutions can more mildly approximate the ideal contrastive loss compared with the sifted one. We proceed by describing the first progressive solution.

Ordinary labels support positive feedback for the given class, while complementary labels provide negative feedback for the given class. Thus, [4, 10, 21] solve CLL problems by minimizing predictive

probabilities of complementary labels during the training process. Motivated by this, we introduce a soft way based on the probabilities of complementary labels to correct sampling bias, where the probabilities of complementary labels are defined as $\mathbf{1} - \boldsymbol{f}(\boldsymbol{x})$ according to [10, 21]. The soft contrastive loss is expressed as

$$\bar{L}_{\text{soft}} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p, \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+, \boldsymbol{x}_i^-\sim p} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \sum_{i=1}^{N} (1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}_i^-)) e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}} \right], \tag{9}$$

where $f_{h(\boldsymbol{x})}(\boldsymbol{x}_i^-)$ denotes $h(\boldsymbol{x})$th element of $\boldsymbol{f}(\boldsymbol{x}_i^-)$. Intuitively, $1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}_i^-)$ refers to the predictive probability of the label $h(\boldsymbol{x})$ being the complementary label of $\boldsymbol{x}_i^-$. If $1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}_i^-)$ is higher, the predictive label of $\boldsymbol{x}$ is a complementary label of $\boldsymbol{x}_i^-$ with a higher possibility. Then, it is easy to derive $\boldsymbol{x}_i^-$ is a truly negative of $\boldsymbol{x}$ when $h(\boldsymbol{x}) = \arg\max_j (1 - f_j(\boldsymbol{x}_i^-))$, because the predictive label $h(\boldsymbol{x})$ of $\boldsymbol{x}$ belongs to the complementary labels of $\boldsymbol{x}_i^-$ while the complementary label is an incorrect label of an instance. Moreover, predictive probabilities of complementary labels are used to dynamically weight negatives, which avoids unexpected ambiguities of the sifted contrastive loss caused by roughly deleting negatives.

Similar to the sifted contrastive loss $\bar{L}_{\text{sifted}}$, we establish a bound of the soft contrastive loss $\bar{L}_{\text{soft}}$ to theoretically analyze its feasibility in approximating the ideal contrastive loss. The established bound of $\bar{L}_{\text{soft}}$ is shown in the theorem below.

Theorem 4. *There exists a parameter assignment θ that makes $\boldsymbol{f}$ be a perfect classifier, such that for any $\boldsymbol{x}$, $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$. Then h can assign the ground-truth label for $\boldsymbol{x}$. With any embedding function $\boldsymbol{g}$ and $N \to \infty$, the following inequality holds*

$$L_{\text{ideal}} \le \bar{L}_{\text{soft}} \le L_{\text{std}}.$$

The proof is stated in Appendix C. According to Theorem 4, the ideal contrastive loss L_{ideal} and the standard contrastive loss L_{std} are the lower and upper bounds for our soft contrastive loss $\bar{L}_{\text{soft}}$, respectively. Compared with L_{std}, $\bar{L}_{\text{soft}}$ shows a tighter upper bound on the ideal contrastive loss L_{ideal}. This observation illustrates the feasibility of $\bar{L}_{\text{soft}}$ to approximate the ideal object and explains why $\bar{L}_{\text{soft}}$ exhibits better performance than L_{std} in MCLL.

The second progressive solution is similar to the soft contrastive loss $\bar{L}_{\text{soft}}$, which corrects sampling bias by leveraging the property that minimizes the predictive probabilities of complementary labels to dynamically weight per negative instance. Differ from $\bar{L}_{\text{soft}}$, the generation of dynamic weights depends on MCLs $\bar{Y}$ of $\boldsymbol{x}$ rather than the predictive label of $\boldsymbol{x}$, hence, we define the weighted contrastive loss as

$$\bar{L}_{\text{wtd}} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p, \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+, \boldsymbol{x}_i^-\sim p} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \sum_{i=1}^{N} \bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}_i^-) e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}} \right]. \tag{10}$$

It is obvious that $\bar{Y}$ must not contain the ground-truth label of $\boldsymbol{x}$, so the instance $\boldsymbol{x}_i^-$ has a chance to be a true negative instance of $\boldsymbol{x}$ when $h(\boldsymbol{x}_i^-) \in \bar{Y}$. Let us explain the setting of $\bar{L}_{\text{wtd}}$ with an MCLL example of five labels ($K = 5$). Suppose $\bar{\boldsymbol{y}}^T = [1, 1, 0, 1, 0]$ denotes the MCLs of $\boldsymbol{x}$ and the ground-truth label of $\boldsymbol{x}$ is 3. We assume $\boldsymbol{x}_i^-$ is a true negative instance of $\boldsymbol{x}$ and $\boldsymbol{f}(\boldsymbol{x}_i^-)^T = [0.1, 0.5, 0.1, 0.2, 0.1]$ denotes the predicted posterior probability of the five labels for $\boldsymbol{x}_i^-$. According to Equation (10), we have $\bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}_i^-) = 0.8$. Similarly, if $\boldsymbol{x}_i^-$ is a false negative instance of $\boldsymbol{x}$ and $\boldsymbol{f}(\boldsymbol{x}_i^-)^T = [0.1, 0.1, 0.6, 0.1, 0.1]$, then $\bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}_i^-) = 0.3$. In this way, the weight of the true negative instance is higher than the false one, which naturally leads to $\bar{L}_{\text{wtd}}$ can mildly approximate the ideal loss $\bar{L}_{\text{ideal}}$ and achieve the goal of alleviating sampling bias.

Similarly, we illustrate the feasibility of $\bar{L}_{\text{wtd}}$ theoretically by Theorem 5.

Theorem 5. *There exists a parameter assignment θ to enable $\boldsymbol{f}$ to be a perfect classifier, such that for any $\boldsymbol{x}$, $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$. With any embedding function $\boldsymbol{g}$ and $N \to \infty$, it holds that*

$$L_{\text{ideal}} \leq \bar{L}_{\text{wtd}} \leq L_{\text{std}}.$$

The proof is presented in Appendix D. According to Theorem 5, $\bar{L}_{\text{wtd}}$ lies between the ideal contrastive loss and the standard contrastive loss, which proves the feasibility of $\bar{L}_{\text{wtd}}$ in addressing the MCLL problem. As discussed earlier, both $\bar{L}_{\text{soft}}$ and $\bar{L}_{\text{wtd}}$ represent progressive solutions to mildly approximate L_{ideal} while avoid unexpected ambiguities caused by roughly deleting negatives in $\bar{L}_{\text{sifted}}$.

4.4 Difference between $\bar{L}_{\text{soft}}$ and $\bar{L}_{\text{wtd}}$

As discussed above, the two progressive solutions, $\bar{L}_{\text{soft}}$ and $\bar{L}_{\text{wtd}}$, can alleviate the sampling bias problem and approximate the ideal contrastive loss more mildly than the sifted one. Naturally, a question arises: what is the difference between these two progressive solutions? Therefore, we will explore their differences in this subsection. The primary distinction between the loss functions $\bar{L}_{\text{soft}}$ and $\bar{L}_{\text{wtd}}$ lies in their approaches to weighting negative samples within the denominator of their respective formulas. These methods are not only mathematically distinct but also address different challenges in learning dynamics.

$\bar{L}_{\text{soft}}$ dynamically weights negatives based on the probabilities that these negatives do not match the predictive label of $\boldsymbol{x}$, denoted as $(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}_i^-))$. This method effectively adjusts the impact of each negative instance based on its likelihood of being a true negative, given the current model's predictions. High confidence in the predicted label leads to lower weights for corresponding negatives, reducing their impact on the loss and focusing the model's learning on more ambiguous instances. In contrast, $\bar{L}_{\text{wtd}}$ incorporates a broader spectrum of information by weighting each negative instance using $\bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}_i^-)$, which reflects the properties of MCLs. This method not only addresses the individual likelihood of each label being incorrect but also considers the relationships among various possible labels. This richer contextual basis can make $\bar{L}$wtd more effective in environments where the label space is complex or where the classifier's predictions are less reliable. By taking into account a wider range of label interactions, $\bar{L}_{\text{wtd}}$ may enhance the robustness of the learned representations, especially in diverse and challenging datasets.

While both loss functions serve as contrastive mechanisms designed through the careful weighting of negative samples, they shine under slightly different circumstances. $\bar{L}_{\text{soft}}$ is particularly useful in early training phases or simpler tasks where model confidence is a reliable indicator of correctness. On the other hand, $\bar{L}_{\text{wtd}}$ could be more advantageous in later stages or complex scenarios where a deeper understanding of the label relationships and a broader set of negatives can lead to better generalization. The choice between these two progressive solutions may depend on the specific characteristics of the dataset and the robustness requirements of the application at hand. Specifically, the sifted contrastive loss is suitable when predictions are relatively reliable, while the soft and weighted variants provide more robust supervision when predictions are noisy. Subsequently, the objective of MulCo combined with the soft contrastive loss $\bar{L}_{\text{soft}}$ is expressed as

$$\bar{\mathcal{L}}_{\text{soft}} = \bar{L}_{\log} + \lambda \bar{L}_{\text{soft}}. \tag{11}$$

MulCo with the weighted contrastive loss $\bar{L}_{\text{wtd}}$ are defined as

$$\bar{\mathcal{L}}_{\text{wtd}} = \bar{L}_{\log} + \lambda \bar{L}_{\text{wtd}}. \tag{12}$$

Algorithm 1: MulCo

Input:
$\bar{\mathcal{D}}$: the multiple-complementary-label training set;
E: the number of epochs;
m: the momentum coefficient;
Output:
θ: encoder parameter of $g(\cdot)$;
θ': momentum encoder parameter of $g'(\cdot)$;
$\boldsymbol{f}(\boldsymbol{x})$: the predictive probability of $p(\boldsymbol{y}|\boldsymbol{x})$;
$h(\boldsymbol{x})$: assigning a label for $\boldsymbol{x}$;

Let $\bar{\mathcal{L}}$ be the training loss;
if $\bar{\mathcal{L}}_{\text{sifted}}$ *is applied* **then**
  $\bar{\mathcal{L}} = \bar{\mathcal{L}}_{\text{sifted}}$;
else if $\bar{\mathcal{L}}_{\text{soft}}$ *is applied* **then**
  $\bar{\mathcal{L}} = \bar{\mathcal{L}}_{\text{soft}}$;
else if $\bar{\mathcal{L}}_{\text{wtd}}$ *is applied* **then**
  $\bar{\mathcal{L}} = \bar{\mathcal{L}}_{\text{wtd}}$;
for *epoch in* E **do**
  for $\boldsymbol{x}$ *in* Dataloader($\bar{\mathcal{D}}$) **do**
    $\boldsymbol{q} = g(\boldsymbol{x})$;
    $\boldsymbol{k} = g'(\boldsymbol{x})$;
    $\bar{\mathcal{L}}(\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{f}(\boldsymbol{x}), h(\boldsymbol{x}))$;
    $\theta = \text{SGD}(\bar{\mathcal{L}}, \theta)$;
  end
  $\theta' = m * \theta' + (1 - m) * \theta$;
end

The overall procedure of the proposed method is shown in Algorithm 1. It is worth noting that only one contrastive loss is used throughout the training phase, while three contrastive losses are presented as alternative designs to address sampling bias in different ways.

5 Experiments

In this section, we evaluate the proposed framework with comparative studies against SOTA baselines. The implementation is based on PyTorch [26] and NVIDIA TITAN RTX. The code will be released upon acceptance.

5.1 Experimental Settings

Datasets. Three widely used datasets KMNIST [6], CIFAR-10, and CIFAR-100 [22], and three datasets (Yeast, Texture, and Dermatology) from the UCI repository[1] are applied to experimental studies. KMNIST dataset consists of 60,000 training instances and 10,000 test instances, distributed across 10 classes. CIFAR-10 dataset and CIFAR-100 consist of 60,000 color images, which are split into 50,000 training images and 10,000 test images. CIFAR-10 has 10 classes, and CIFAR-100 has 100 classes. Three datasets from the UCI repository are regular-scale datasets, where we divide the dataset into training and test sets at a 9:1 ratio. The generation of MCLs follows [7, 35], let s ($s \in \{1, 2, \ldots, K-1\}$) be the number of complementary labels for each instance $\boldsymbol{x}$, then we uniformly and randomly sample a complementary label set $\bar{Y}$ with size s per instance $\boldsymbol{x}$. The training data is annotated by complementary labels, while the test set with ordinary labels is used to evaluate the performance

[1]The datasets can be available at https://archive.ics.uci.edu.

of all methods. For CIFAR-10 and CIFAR-100, we both adopt normalization, horizontal flipping, and random cropping as data augmentation ways.

Base Model. Our architecture is similar to MoCo [13] other than the objective loss, where the MLP ($d - 300 - K$) model and the 18-layer ResNet [14] are used as the backbone for representation learning of KMNIST and CIFAR datasets, respectively. We use a linear model as the backbone for the three datasets from the UCI repository. Most experimental setups for the contrastive network follow previous works [13, 37], where the projection head of the contrastive network is a 2-layer MLP with 128-dimensional embedding output. The size of the queue used to store key (negatives) embeddings is fixed to 8,192, and the momentum coefficient is set as 0.999 for a momentum encoder network updating.

Baselines. We employ four SOTA CLL methods to be compared, including UB-EXP, UB-LOG (Equation (1)) [7], L-UW [10], MLCL [9], ComCoComCo [18] and PLNL [25]. The above methods solve the CLL problem by designing different loss functions. Following Feng et al. [7], we use MSE loss and **Generalized Cross Entropy (GCE)** loss [48] as baselines, which are robust to noisy labels. In addition, a popular PLL method PiCO [36], and a self-supervised learning method [5] are both adopted as baselines in this article. We briefly introduce all comparison methods below.

- *SCLL method*: L-UW [10] is proposed for SCLL in multi-class classification, while MLCL [9] is designed for the multi-label classification task. Since the SCLL problem is a special case of MCLL, we extend both methods to learn with MCLs. Moreover, we adopt the softmax function instead of the sigmoid function to better adapt MLCL to the MCLL setting.
- *MCLL method*: Four comparison methods, UB-EXP, UB-LOG, ComCo [18], and PLNL [25] are designed for the MCLL problem. UB-EXP and UB-LOG are bounded losses (UB means the short form of *upper bounded*), which are both designed by Feng et al. [7]. ComCo introduces a contrastive learning framework for MCLs by leveraging instance-level discrimination to alleviate the ambiguity [18]. PLNL is a pseudo-labeling-based approach that leverages confidence estimation to infer candidate positive labels from complementary supervision [25].
- *Loss variant*: GCE is proposed by Zhang and Sabuncu [48] to deal with the noisy label learning problem, and MSE loss has been proven to be a robust loss for noisy label learning [12]. Obviously, GCE and MSE losses are robust to data with noisy labels. Therefore, we modify GCE and MSE losses to make them available for MCLL according to Feng et al. [7].
- *PLL method*: PiCO is a representative PLL method [36, 37], where each instance is equipped with multiple candidate class labels. Here, PiCO deals with MCLL problem by regarding $\mathcal{Y} \setminus \bar{Y}$ as candidate labels.
- *Self-supervised learning method*: Chuang et al. [5] propose a debiased contrastive loss to alleviate the problem of sampling bias in contrastive learning. We apply the framework of MulCo with the loss $\bar{L}_{\log}$ and the debiased contrastive loss to verify the effectiveness of our method in correcting the sampling bias problem.

Dynamic Tradeoff Parameter λ. The objective presented in Section 4 is a weighted combination of the MCLL loss and improved contrastive loss functions, which is controlled by a hyperparameter λ. In this article, we leverage the dynamic tradeoff parameter λ proposed by Wang et al. [35]. Here, we will illustrate the principle and advantage of the dynamic tradeoff parameter λ. The predictions of the classifier in the early stage are highly random, which will produce low-quality predictions [35]. Hence, if the value of λ is large in the beginning epochs, the modified contrastive loss functions using low-quality predictions to correct sampling bias would cause error accumulation during training. This motivates us to use a small value in the early stage during the training process and a relatively large one in the later stage. To achieve the above target, we apply the trick of the dynamic

tradeoff parameter proposed by Wang et al. [35], which is defined as

$$\Lambda(t) = \min\left\{\frac{t}{E'}\lambda, \lambda\right\}, \tag{13}$$

where t denotes tth epoch. The fixed tradeoff parameter λ in Equations (6), (11), and (12) is replaced by Equation (13), our method trains with a dynamic tradeoff parameter that is 0 at the beginning epoch and increases to λ at $t = E'$. It keeps a constant λ after E'th epoch until the end of the training. Subsequently, we will discuss the impact of the fixed tradeoff parameter and the dynamic tradeoff parameter on the performance of our method.

Setup. Stochastic gradient descent [27] with momentum 0.9 is applied for our optimization. Weight decay for CIFAR-10 is set as 10^{-3}, while that for three datasets of UCI, KMNIST and CIFAR-100 are 10^{-4}. The learning rate is selected from $\{0.1, 0.05, 0.01, 0.001, 0.005\}$ and batch-size set to 64. We train for 200 epochs with the learning rate multiplied by 0.1 at 100 and 150 epochs [42]. Following [5], the temperature scaling is set as $\tau = 0.05$. The hyperparameters used in Equation (13) are set as $E' = 100$ and $\lambda = 1$. For a fair comparison, we adopt the same model, data augmentation, optimizer, learning rate, weight decay, and learning policies for all methods as our method. PiCO still maintains the strategy of warm-up and data augmentation reported in the original literature to achieve comparable results.

5.2 Main Empirical Results

We show the mean and std of test accuracy in Tables 1–3. MulCo with debiased refers to that we use the debiased contrastive loss and $\bar{L}_{\log}$ to learn. As shown in Tables 1 and 2, our strategies of correcting the sampling bias problem perform better than the debiased contrastive loss, which demonstrates that our strategies are suitable for MCLL. Moreover, we observe that all algorithms consistently have better results as the number of complementary labels increases in the CIFAR-10 dataset, especially for PiCO, whose performance significantly improves at $s = 3$ compared to $s = 2$. This is because the supervised information increases as the number of complementary labels increases. The pseudo target of PiCO relies on the predictive probability, where the pseudo target quality is low when predictions of the method are low quality, caused by the weaker supervised information.

The task of CIFAR-100 is a challenge compared with CIFAR-10, while our method significantly outperforms all baselines in $s \in \{40, 50, 60\}$. The quality of predictive probabilities of methods becomes lower as the task becomes harder, which naturally results in a dissatisfactory performance for methods that depend on predictions to design improved strategies, such as PiCO. Though our method has a robust performance on CIFAR-100 than baselines, this phenomenon is also slightly reflected in our proposed method: MulCo with $\bar{\mathcal{L}}_{\text{soft}}$ or $\bar{\mathcal{L}}_{\text{wtd}}$ is superior to that with $\bar{\mathcal{L}}_{\text{sifted}}$ when s keeps a lower size. When $s = 30$, the complementary supervision is relatively weak, under which ComCo achieves comparable performance. As more complementary labels become available, MulCo can better exploit the increased supervision by correcting sampling bias, leading to superior performance.

In addition, we evaluate MulCo on regular-scale datasets from the UCI repository to assess its robustness across different domains, where s is set to half of the label space size for each dataset. As shown in Table 3, MulCo demonstrates robust performance across all three UCI datasets, with its contrastive loss variants matching or outperforming SOTA baselines. Although our method performs slightly worse on Yeast, it remains competitive and closely aligned with the best baseline. Moreover, MulCo achieves highly competitive results on Texture and Dermatology, demonstrating its effectiveness in mitigating sampling bias via contrastive learning and learning more discriminative representations for the MCLL problem.

Table 1. Test Accuracy (Mean ± Std, %) of Each Algorithm for Five Trials

Datasets	Methods	$s = 2$	$s = 3$	$s = 4$	$s = 5$
CIFAR-10	UB-EXP	86.39 ± 0.24•	89.35 ± 0.28•	91.06 ± 0.11•	92.23 ± 0.12•
	UB-LOG	**87.45 ± 0.30**	90.36 ± 0.27	91.99 ± 0.31	92.87 ± 0.17
	MSE	74.03 ± 0.47•	81.10 ± 0.35•	86.23 ± 0.24•	89.52 ± 0.34•
	GCE	71.45 ± 0.64•	79.81 ± 0.29•	86.18 ± 0.50•	90.18 ± 0.12•
	L-UW	85.72 ± 0.19•	88.79 ± 0.08•	90.68 ± 0.27•	92.15 ± 0.13•
	MLCL	85.62 ± 0.26•	88.33 ± 0.26•	90.35 ± 0.12•	91.65 ± 0.07•
	PiCO	52.71 ± 7.24•	83.76 ± 7.05•	92.05 ± 0.10	92.95 ± 0.06
	ComCo	66.28 ± 0.65•	70.12 ± 0.58•	73.40 ± 0.91•	75.89 ± 0.40•
	PLNL	85.82 ± 0.22•	88.72 ± 0.21•	89.74 ± 0.48•	90.92 ± 0.07•
	MulCo with debiased	86.16 ± 0.12•	89.41 ± 0.33•	91.63 ± 0.29•	92.73 ± 0.15
	MulCo with $\bar{\mathcal{L}}_{\text{sifted}}$	87.17 ± 0.18	90.32 ± 0.19	91.96 ± 0.19	**93.02 ± 0.15**
	MulCo with $\bar{\mathcal{L}}_{\text{soft}}$	87.33 ± 0.33	90.26 ± 0.37	**92.16 ± 0.23**	92.98 ± 0.19
	MulCo with $\bar{\mathcal{L}}_{\text{wtd}}$	87.24 ± 0.40	**90.36 ± 0.29**	92.02 ± 0.34	92.97 ± 0.20
Datasets	Methods	$s = 30$	$s = 40$	$s = 50$	$s = 60$
CIFAR-100	UB-EXP	13.07 ± 2.96•	19.81 ± 1.74•	32.26 ± 3.25•	38.54 ± 3.26•
	UB-LOG	17.13 ± 1.62•	28.18 ± 1.50•	47.18 ± 3.16•	60.12 ± 1.15•
	MSE	15.81 ± 0.93•	26.50 ± 1.69•	44.51 ± 1.78•	56.06 ± 0.71•
	GCE	09.26 ± 0.42•	13.55 ± 0.58•	20.41 ± 0.92•	33.60 ± 0.84•
	L-UW	12.47 ± 1.09•	25.17 ± 1.66•	43.41 ± 1.58•	58.28 ± 1.45•
	MLCL	13.12 ± 1.63•	26.80 ± 2.38•	45.72 ± 1.75•	59.06 ± 0.98•
	PiCO	12.93 ± 0.73•	13.26 ± 0.96•	15.90 ± 1.67•	18.53 ± 1.07•
	ComCo	**23.89 ± 0.18**○	35.14 ± 0.32•	42.86 ± 0.32•	50.36 ± 0.42•
	PLNL	13.44 ± 0.21•	18.04 ± 0.62•	32.91 ± 0.64•	51.78 ± 0.76•
	MulCo with debiased	18.75 ± 0.55•	35.68 ± 2.65•	53.22 ± 1.46•	62.83 ± 0.39•
	MulCo with $\bar{\mathcal{L}}_{\text{sifted}}$	19.01 ± 1.06	37.42 ± 4.19	56.06 ± 1.01	63.02 ± 0.52
	MulCo with $\bar{\mathcal{L}}_{\text{soft}}$	20.36 ± 0.49	37.14 ± 2.61	55.92 ± 1.36	63.36 ± 0.33
	MulCo with $\bar{\mathcal{L}}_{\text{wtd}}$	20.31 ± 0.75	**38.56 ± 2.99**	**56.18 ± 0.72**	**63.38 ± 0.45**

The best performance is presented in boldface, where •/○ indicates whether the performance of our method (the best of $\bar{\mathcal{L}}_{\text{sifted}}$, $\bar{\mathcal{L}}_{\text{soft}}$, and $\bar{\mathcal{L}}_{\text{wtd}}$) is superior/inferior to baselines (with 5% paired t-test). s presents the number of complementary labels per training instance.

Table 2. Test Accuracy (Mean ± Std, %) of All Methods on KMNIST over Five Trials

s	UB-EXP	UB-LOG	MSE	GCE	L-UW	MLCL	PiCO	MulCo with			
								Debiased	$\bar{\mathcal{L}}_{\text{sifted}}$	$\bar{\mathcal{L}}_{\text{soft}}$	$\bar{\mathcal{L}}_{\text{wtd}}$
2	58.33 ± 1.39•	59.36 ± 1.00•	54.39 ± 5.15•	53.14 ± 6.64•	55.14 ± 1.64•	45.32 ± 2.19•	51.59 ± 0.65•	51.69 ± 1.49•	62.63 ± 0.77	**62.69 ± 0.85**	62.67 ± 0.82
3	60.45 ± 0.49•	61.53 ± 1.05•	63.07 ± 0.49•	60.94 ± 0.93•	58.31 ± 0.26•	50.52 ± 0.61•	54.25 ± 0.46•	54.25 ± 1.14•	66.05 ± 1.43	65.97 ± 1.47	**66.07 ± 1.47**
4	61.51 ± 0.30•	64.24 ± 0.56•	66.59 ± 0.55	65.01 ± 0.50•	60.53 ± 0.19•	54.41 ± 1.29•	58.75 ± 0.53•	57.13 ± 1.31•	68.78 ± 1.39	**68.83 ± 1.43**	68.81 ± 1.40
5	61.94 ± 0.24•	66.47 ± 0.69•	69.17 ± 0.46•	68.16 ± 0.33•	61.78 ± 0.15•	55.13 ± 2.24•	59.77 ± 0.39•	58.39 ± 1.25•	70.56 ± 0.76	**70.59 ± 0.82**	70.55 ± 0.74

The best performance is presented in boldface, where •/○ indicates whether the performance of our method is superior/inferior to baselines (with 5% paired t-test).

5.3 Additional Experiments

Effect of Correcting Sampling Bias Strategy. We ablate the contributions of two key components of MulCo: contrastive learning and sampling bias correcting. Specifically, we compare MulCo with two variants: (1) *MulCo w/o contrastive term*, which removes the contrastive learning module and trains the model solely with the MCLL loss in Equation (2), without the dynamic tradeoff parameter; (2) *MulCo with* $\mathcal{L}_{\text{std}}$, where the contrastive term is replaced by a standard contrastive

Table 3. Test Accuracy (Mean ± Std, in %) of All Methods on Three Datasets from the UCI Repository over Three Trials

Datasets	UB-EXP	UB-LOG	MSE	GCE	L-UW	MLCL	PiCO	ComCo	PLNL	MulCo with			
										Debiased	$\bar{\mathcal{L}}_{\text{sifted}}$	$\bar{\mathcal{L}}_{\text{soft}}$	$\bar{\mathcal{L}}_{\text{wtd}}$
Yeast	56.60• (±1.47)	57.64 (±1.91)	**58.17**○ (±1.84)	57.29 (±0.62)	57.65 (±1.86)	56.07• (±1.46)	52.90• (±1.20)	36.12• (±11.99)	44.97• (±0.55)	56.79 (±1.12)	57.20 (±1.47)	56.79 (±1.12)	56.16 (±2.00)
Texture	93.62• (±0.37)	93.62• (±0.92)	92.71• (±0.49)	90.10• (±1.33)	93.75 (±0.55)	93.75 (±0.32)	93.15• (±0.21)	67.71• (±3.59)	26.51• (±3.97)	94.34 (±1.17)	**94.64** (±1.26)	94.49 (±1.05)	94.20 (±0.63)
Dermatology	95.49• (±3.37)	97.30• (±2.21)	94.59• (±2.21)	95.50• (±2.55)	95.49• (±3.37)	95.49• (±3.37)	98.44 (±1.27)	97.30• (±2.21)	79.28• (±4.59)	97.92• (±1.95)	97.92 (±0.74)	**98.44** (±1.27)	97.92 (±0.74)

The best results are shown in bold, where •/○ indicates whether the performance of our method (the best of $\bar{\mathcal{L}}_{\text{sifted}}$, $\bar{\mathcal{L}}_{\text{soft}}$, and $\bar{\mathcal{L}}_{\text{wtd}}$) is superior/inferior to baselines (with 5% paired t-test). The number of MCLs per instance is set to half of the label space size for each dataset.

Table 4. Ablation Study on CIFAR-10 and CIFAR-100 (in %)

	CIFAR-10				CIFAR-100			
	$s = 2$	$s = 3$	$s = 4$	$s = 5$	$s = 30$	$s = 40$	$s = 50$	$s = 60$
MulCo w/o contrastive term	86.97	90.18	91.81	92.80	16.66	26.54	44.44	58.92
MulCo with $\mathcal{L}_{\text{std}}$	86.49	90.01	91.77	92.95	18.57	34.61	54.64	62.38
MulCo with $\bar{\mathcal{L}}_{\text{sifted}}$	87.34	90.62	92.23	**93.22**	19.63	36.92	56.48	63.06
MulCo with $\bar{\mathcal{L}}_{\text{soft}}$	87.35	**90.83**	92.29	92.69	20.62	37.89	56.89	63.28
MulCo with $\bar{\mathcal{L}}_{\text{wtd}}$	**87.51**	90.74	**92.33**	93.12	**20.74**	**38.01**	**56.92**	**63.69**

The best performance is presented in boldface, where s presents the number of complementary labels per training instance.

loss without sampling bias correction, i.e., the objective is $\mathcal{L}\text{std} = \bar{L}\log + \lambda L_{\text{std}}$, and λ follows the same dynamic tradeoff strategy as in our method. All other experimental settings, including base models and hyperparameters, are kept identical to those of MulCo across all datasets, as described in Section 5.1.

From Table 4, our method outperforms two variants on CIFAR-10 and CIFAR-100. MulCo with $\bar{\mathcal{L}}_{\text{wtd}}$ is 4.08%, 11.47%, 12.48%, and 4.77% higher than variant (1) on CIFAR-100 with 30, 40, 50, and 60 complementary labels for each instance, and MulCo with $\mathcal{L}_{\text{std}}$ also achieves comparable or superior performance than variant (1) on all settings. This demonstrates that contrastive learning can promote MCLL to distinguish the direction of representation learning, which is important for producing better representations. In addition, MulCo with $\bar{\mathcal{L}}_{\text{sifted}}$, $\bar{\mathcal{L}}_{\text{soft}}$ and $\bar{\mathcal{L}}_{\text{wtd}}$ is superior to variant (2), which indicates sampling negative instance from truly different labels improves performance and our method corrects sampling bias effectively. We further observe that the performance gains on CIFAR-10 are relatively smaller than those on CIFAR-100. CIFAR-10 has a smaller label space with lower ambiguity, under which variant (1) already provides relatively informative supervision, leaving limited room for further improvement. In contrast, CIFAR-100 involves a larger number of classes with higher ambiguity, where correcting sampling bias and enhancing representation discrimination become more critical, making the advantages of MulCo more evident.

Relationship between Performance and the Number of Complementary Labels. Figure 2 illustrates the performance of five methods with different sizes of s that denotes the number of complementary labels for each training instance. Although PiCO shows a good performance in PLL, PiCO still fails to solve the MCLL problem according to the evidence of the huge gap between PiCO and our method

in the performance (as shown in Figure 2). Obviously, MulCo uses the improved contrastive losses $\bar{\mathcal{L}}_{\text{sifted}}$, $\bar{\mathcal{L}}_{\text{soft}}$ and $\bar{\mathcal{L}}_{\text{wtd}}$ to achieve superior results compared MulCo with the standard contrastive loss $\mathcal{L}_{\text{std}}$, which demonstrates alleviating the problem of sampling bias is useful to further improve contrastive learning performance of MulCo. Moreover, the performance of all methods improves as s increases, which is caused by the stronger supervised information. We observe that MulCo finally achieves results that are comparable to the *fully supervised contrastive learning model*, which indicates that contrastive learning can sufficiently distinguish the representation learning direction. The comparison highlights the superiority of our method, which simultaneously shows the application prospect of our method.

Performance of MulCo with Different λ. We then explore the effect of varying λ values that tradeoff the UB-LOG (typical MCLL loss) and contrastive losses on MulCo performance. Table 5 reports results of MulCo with different λ values, where "Fixed" presents the value of λ keeps invariably from the beginning to the end of the training, while "Dynamic" denotes λ uses the dynamic tradeoff parameter strategy (shown in Equation (13)). We can observe that the best performance is obtained at dynamic λ on CIFAR-10 and CIFAR-100. From the results of the fixed λ ($\lambda \in \{0.1, 0.2, 0.5, 1\}$) shown, a relatively small λ usually causes a good performance than a larger value, which demonstrates the contrastive network tends to fit low-quality predictions at the early stage of training when λ is large. In general, MulCo performs well when the dynamic tradeoff parameter strategy is applied for λ.

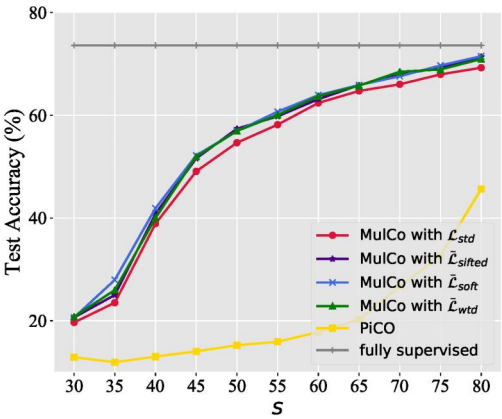


Fig. 2. Performance of different methods on CIFAR-100 with various s, where s presents the number of complementary labels per training instance. *Fully supervised* in the figure denotes the fully supervised contrastive learning model.

Execution Time. In Table 6, we present the running time of each method on the three datasets of the UCI repository and the CIFAR-10 datasets (the running time of CIFAR-100 is similar to that of CIFAR-10, hence only providing CIFAR-10's results). Shorter execution times generally indicate lower computational complexity of the method. It can be observed from Table 6 that MulCo with the three proposed variants ($\bar{\mathcal{L}}_{\text{sifted}}$, $\bar{\mathcal{L}}_{\text{soft}}$, and $\bar{\mathcal{L}}_{\text{wtd}}$) achieves running times comparable to existing contrastive-based methods such as PiCO and ComCo, with only marginal differences across datasets. The three variants exhibit similar computational costs, suggesting that the proposed sampling bias correction and dynamic weighting strategies introduce negligible overhead. Overall, these results show that MulCo improves performance without substantially increasing computational or model complexity, demonstrating the feasibility of the proposed framework.

5.4 Visualization of Learned Representation

We visualize the representation learned by the feature embedding function through t-SNE [34] in Figure 3. Different colors present different correct class labels. We contrast the t-SNE embeddings of six methods, including PiCO, UB-LOG, MulCo with $\mathcal{L}_{\text{std}}$, $\bar{\mathcal{L}}_{\text{sifted}}$, $\bar{\mathcal{L}}_{\text{soft}}$, $\bar{\mathcal{L}}_{\text{wtd}}$, on the CIFAR-10 dataset with $s = 2$ and CIFAR-100 dataset with $s = 50$. For CIFAR-100, we visualize the learned representation for instances from the first 10 classes. We can observe that the representation of PiCO is indistinguishable since its pseudo target suffers from high uncertainty caused by low-quality predictions. Figure 3(a) and (b) prove this of PiCO, the prediction quality is worse, and the

Table 5. Test Accuracy of MulCo with Different λ on CIFAR-10 and CIFAR-100 (in %)

MulCo with	CIFAR-10, $s = 2$			CIFAR-100, $s = 30$			CIFAR-100, $s = 40$		
	$\bar{\mathcal{L}}_{\text{sifted}}$	$\bar{\mathcal{L}}_{\text{soft}}$	$\bar{\mathcal{L}}_{\text{wtd}}$	$\bar{\mathcal{L}}_{\text{sifted}}$	$\bar{\mathcal{L}}_{\text{soft}}$	$\bar{\mathcal{L}}_{\text{wtd}}$	$\bar{\mathcal{L}}_{\text{sifted}}$	$\bar{\mathcal{L}}_{\text{soft}}$	$\bar{\mathcal{L}}_{\text{wtd}}$
Fixed tradeoff parameter λ									
$\lambda = 0.1$	87.07	**87.44**	87.07	17.52	20.62	20.30	35.83	36.26	36.20
$\lambda = 0.2$	87.11	87.08	86.94	20.50	16.33	20.62	40.13	37.15	37.04
$\lambda = 0.5$	87.20	87.22	87.38	19.04	19.52	19.52	38.85	37.08	37.36
$\lambda = 1$	87.32	87.28	87.18	17.21	19.50	16.88	36.05	33.18	35.74
Dynamic tradeoff parameter λ									
$\lambda = 1$	**87.34**	87.35	**87.51**	**20.65**	**20.62**	**20.78**	**40.84**	**41.88**	**40.00**

The best performance is presented in boldface, where s presents the number of complementary labels per training instance.

Table 6. Running Time (in 10^2 Seconds) of All Methods over 200 Epochs

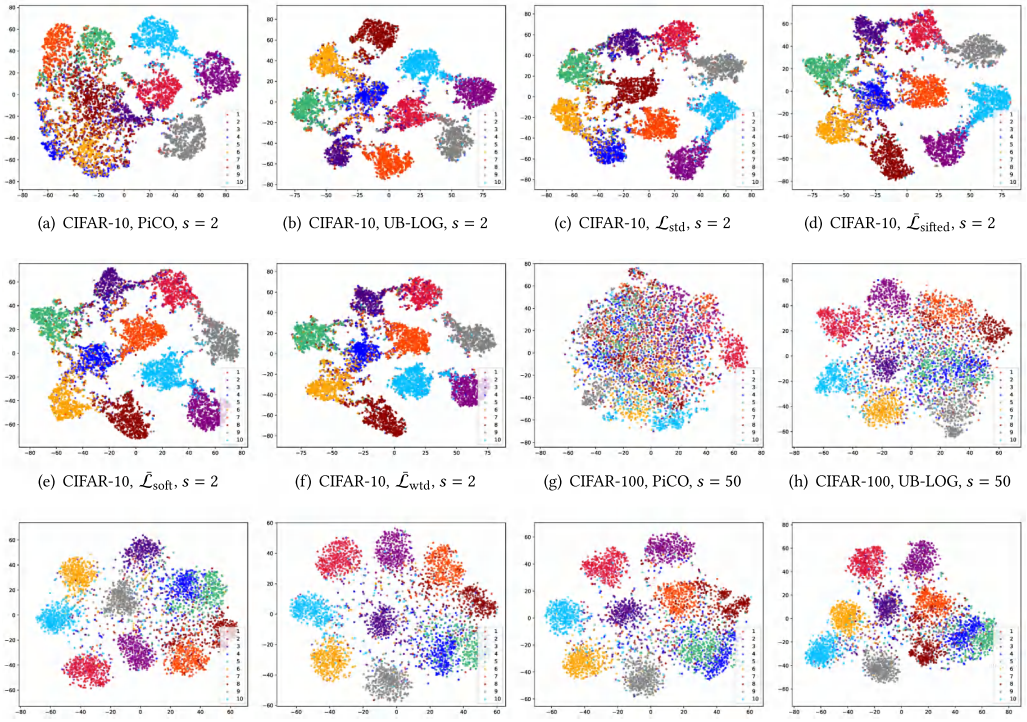
Datasets	UB-EXP	UB-LOG	MSE	GCE	L-UW	MLCL	PiCO	ComCo	PLNL	MulCo with			
										Debiased	$\bar{\mathcal{L}}_{\text{sifted}}$	$\bar{\mathcal{L}}_{\text{soft}}$	$\bar{\mathcal{L}}_{\text{wtd}}$
CIFAR-10	79.51	84.49	79.46	78.87	80.43	78.62	66.23	68.21	97.02	64.64	67.90	65.22	68.17
Yeast	7.17	5.87	7.09	7.84	6.84	7.57	8.16	1.88	19.52	9.05	6.33	9.02	9.45
Texture	8.23	8.82	8.22	8.74	7.90	8.08	9.64	2.79	22.09	10.20	10.75	10.31	10.55
Dermatology	5.90	5.27	6.07	6.10	5.74	6.03	8.38	1.38	13.31	9.05	7.89	9.02	8.94

Lower values indicate better efficiency.

representation of PiCO is more indistinguishable. The representation of typical MCLL loss—UB-LOG—is improved, yet some class overlapping (e.g., blue, green, and gray in CIFAR-100). Obviously, the representation of the method with contrastive learning is better than the former two methods. In contrast, MulCo produces well-separated clusters and more distinguishable representations, which proves the effectiveness of our sampling bias correcting strategies in learning high-quality representations.

6 Conclusion

In this article, we propose a sample yet effective MCLL framework called MulCo, which provides well-separated representations by introducing contrastive learning to alleviate label ambiguities in MCLL. However, we find that the problem of sampling bias will impact the representation effectiveness of contrastive learning because negative instances are uniformly and randomly sampled from the data distribution p, which naturally leads to the sampled negative instance being similar to an anchor $\boldsymbol{x}$. Therefore, we design a sifted contrastive loss for MulCo to correct the sampling of negative pairs with the same label, which theoretically derives that the sifted contrastive loss is an upper bound on the ideal contrastive loss. Besides, to alleviate errors that could be introduced by inexact predictions in the sifted contrastive loss, two progressive solutions depending on the properties of complementary labels are developed to mildly approximate the ideal contrastive loss. The effectiveness of the proposed framework is sufficiently validated with empirical studies over benchmark datasets.



(a) CIFAR-10, PiCO, $s = 2$ (b) CIFAR-10, UB-LOG, $s = 2$ (c) CIFAR-10, $\mathcal{L}_{\text{std}}$, $s = 2$ (d) CIFAR-10, $\bar{\mathcal{L}}_{\text{sifted}}$, $s = 2$

(e) CIFAR-10, $\bar{\mathcal{L}}_{\text{soft}}$, $s = 2$ (f) CIFAR-10, $\bar{\mathcal{L}}_{\text{wtd}}$, $s = 2$ (g) CIFAR-100, PiCO, $s = 50$ (h) CIFAR-100, UB-LOG, $s = 50$

(i) CIFAR-100, $\mathcal{L}_{\text{std}}$, $s = 50$ (j) CIFAR-100, $\bar{\mathcal{L}}_{\text{sifted}}$, $s = 50$ (k) CIFAR-100, $\bar{\mathcal{L}}_{\text{soft}}$, $s = 50$ (l) CIFAR-100, $\bar{\mathcal{L}}_{\text{wtd}}$, $s = 50$

Fig. 3. T-SNE visualization of the representation on CIFAR-10 with $s = 2$ and CIFAR-100 with $s = 50$. (a)–(f) show the representation of CIFAR-10 on various methods and (g)–(l) show the representation of CIFAR-100, where the classes of CIFAR-100 are the first 10 classes. Colors represent the corresponding classes.

Moreover, MCLL finds promising applications in sectors where precise data collection is challenging. One such application scenario is within the medical domain, where healthcare professionals frequently encounter diseases with overlapping symptoms. Medical diagnoses necessitate to precisely identify diseases, which is inherently challenging. MCLL proves invaluable in such contexts, enabling medical experts to confidently exclude certain diseases based on the observed symptoms. By prioritizing the exclusion of the least likely diseases (incorrect labels), the MCLL model can then more accurately infer the most probable conditions, enhancing the diagnostic process. It is noteworthy that the effectiveness of MCLL approaches heavily depends on the quality of complementary labels. In scenarios where complementary labels are noisy, the performance of MCLL approaches may be compromised. Addressing this limitation requires comprehensive solutions, encompassing alternative data generation processes and a tailored loss function. Therefore, we intend to tackle these error-related challenges in our future work.

Acknowledgments

The authors wish to thank the associate editor and anonymous reviewers for their helpful comments and suggestions. We thank the Big Data Center of Southeast University for providing the facility support for the numerical calculations in this article.

References

[1] Yu-Zhou Cao, Shu-Qi Liu, and Yi-Tian Xu. 2022. Multi-complementary and unlabeled learning for arbitrary losses and models. *Pattern Recognition* 124 (2022), 108447.

[2] Nicholas Carlini and Andreas Terzis. 2022. Poisoning and backdooring contrastive learning. In *Proceedings of the 10th International Conference on Learning Representations*.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, 1597–1607.

[4] Yu-Ting Chou, Gang Niu, Hsuan-Tien Lin, and Masashi Sugiyama. 2020. Unbiased risk estimators can mislead: A case study of learning with complementary labels. In *Proceedings of the 37th International Conference on Machine Learning*, 1929–1938.

[5] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. 2020. Debiased contrastive learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS '20)*, 8765–8775.

[6] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. 2018. Deep learning for classical Japanese literature. arXiv:1812.01718. Retrieved from https://arxiv.org/abs/1812.01718

[7] Lei Feng, Takuo Kaneko, Bo Han, Gang Niu, Bo An, and Masashi Sugiyama. 2020. Learning with multiple complementary labels. In *Proceedings of the 37th International Conference on Machine Learning*, 3072–3081.

[8] Yi Gao, Miao Xu, and Min-Ling Zhang. 2023. Unbiased risk estimator to multi-labeled complementary label learning. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence*, 3732–3740.

[9] Yi Gao, Miao Xu, and Min-Ling Zhang. 2024. Complementary to multiple labels: A correlation-aware correction approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, 12 (2024), 9179–9191.

[10] Yi Gao and Min-Ling Zhang. 2021. Discriminative complementary-label learning with weighted loss. In *Proceedings of the 38th International Conference on Machine Learning*, 3587–3597.

[11] Yi Gao, Jing-Yi Zhu, Miao Xu, and Min-Ling Zhang. 2025. Multi-label learning with multiple complementary labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 47, 9 (2025), 8013–8024.

[12] Aritra Ghosh, Himanshu Kumar, and P. S. Sastry. 2017. Robust loss functions under label noise for deep neural networks. In *Proceedings of the 31th AAAI Conference on Artificial Intelligence*, 1919–1925.

[13] Kai-Ming He, Hao-Qi Fan, Yu-Xin Wu, Sai-Ning Xie, and Ross B. Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9726–9735.

[14] Kai-Ming He, Xiang-Yu Zhang, Shao-Qing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

[15] Takashi Ishida, Gang Niu, Wei-Hua Hu, and Masashi Sugiyama. 2017. Learning from complementary labels. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS '17)*, 5639–5649.

[16] Takashi Ishida, Gang Niu, Aditya Krishna Menon, and Masashi Sugiyama. 2019. Complementary-label learning for arbitrary losses and models. In *Proceedings of the 36th International Conference on Machine Learning*, 2971–2980.

[17] Bin-Bin Jia, Jun-Ying Liu, Jun-Yi Hang, and Min-Ling Zhang. 2023. Learning label-specific features for decomposition-based multi-class classification. *Frontiers of Computer Science* 17, 6 (2023), 176348.

[18] Haoran Jiang, Zhihao Sun, and Yingjie Tian. 2024. ComCo: Complementary supervised contrastive learning for complementary label learning. *Neural Networks* 169 (2024), 44–56.

[19] Yasuhiro Katsura and Masato Uchida. 2020. Bridging ordinary-label learning and complementary-label learning. In *Proceedings of the 12th Asian Conference on Machine Learning*, 161–176.

[20] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yong-Long Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS '20)*, 18661–18673.

[21] Youngdong Kim, Jun-Ho Yim, Juseung Yun, and Junmo Kim. 2019. NLNL: Negative learning for noisy labels. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision*, 101–110.

[22] Alex Krizhevsky and Geoffrey Hinton. 2009. Learning multiple layers of features from tiny images. *Handbook of Systemic Autoimmune Diseases* 1, 4 (2009), 1–7.

[23] Jun-Nan Li, Cai-Ming Xiong, and Steven C. H. Hoi. 2021. CoMatch: Semi-supervised learning with contrastive graph regularization. In *Proceeding of the 2021 IEEE International Conference on Computer Vision*, 9475–9484.

[24] Jun-Nan Li, Pan Zhou, Cai-Ming Xiong, and Steven C. H. Hoi. 2021. Prototypical contrastive learning of unsupervised representations. In *Proceedings of the 9th International Conference on Learning Representations*.

[25] Yuhang Li, Zhuying Li, and Yuheng Jia. 2025. Complementary label learning with positive label guessing and negative label enhancement. In *Proceedings of the 13th International Conference on Learning Representations*.

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Ze-Ming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 8024–8035.

[27] Herbert Robbins and Sutton Monro. 1951. *A stochastic approximation method*. *The Annals of Mathematical Statistics* 22, 3 (1951), 400–407.

[28] Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2021. Contrastive learning with hard negative samples. In *Proceedings of the 9th International Conference on Learning Representations*.

[29] Nikunj Saunshi, Orestis Plevrakis, Sanjeev Arora, Mikhail Khodak, and Hrishikesh Khandeparkar. 2019. A theoretical analysis of contrastive unsupervised representation learning. In *Proceedings of the 36th International Conference on Machine Learning*, 5628–5637.

[30] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. 2016. Deep metric learning via lifted structured feature embedding. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, 4004–4012.

[31] Wei Tang, Weijia Zhang, and Min-Ling Zhang. 2024. Multi-instance partial-label learning: Towards exploiting dual inexact supervision. *Science China Information Sciences* 67, 3 (2024), 1–14.

[32] Yong-Long Tian, Dilip Krishnan, and Phillip Isola. 2020. Contrastive multiview coding. In *Proceeding of the 16th European Conference Computer Vision*, 776–794.

[33] Yong-Long Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. 2020. What makes for good views for contrastive learning? In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS '20)*, 6827–6839.

[34] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research* 9 (2008), 2579–2605.

[35] Deng-Bao Wang, Lei Feng, and Min-Ling Zhang. 2021. Learning from complementary labels via partial-output consistency regularization. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence*, 3075–3081.

[36] Haobo Wang, Ruixuan Xiao, Yixuan Li, Lei Feng, Gang Niu, Gang Chen, and Junbo Zhao. 2024. PiCO+: Contrastive label lisambiguation for robust partial label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46, 5 (2024), 3183–3198.

[37] Hao-Bo Wang, Rui-Xuan Xiao, Yi-Xuan Li, Lei Feng, Gang Niu, Gang Chen, and Jun-Bo Zhao. 2022. PiCO: Contrastive label disambiguation for partial label learning. In *Proceedings of the 10th International Conference on Learning Representations*.

[38] Jia-Xing Wang, Yin Zheng, Xiao-Shuang Chen, Jun-Zhou Huang, and Jian Cheng. 2019. Semi-supervised learning with contrastive predicative coding. arXiv:1905.10514. Retrieved from https://arxiv.org/abs/1905.10514

[39] Tong-Zhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, 9929–9939.

[40] Wei-Chien Wang, Euijoon Ahn, Dagan Feng, and Jinman Kim. 2023. A review of predictive and contrastive self-supervised learning for medical images. *Machine Intelligence Research* 20, 4 (2023), 483–513.

[41] Zhi-Fan Wu, Tong Wei, Jian-Wen Jiang, Chao-Jie Mao, Ming-Qian Tang, and Yu-Feng Li. 2021. NGC: A unified framework for learning with open-world noisy data. In *Proceeding of the 2021 IEEE International Conference on Computer Vision*, 62–71.

[42] Zhi-Rong Wu, Yuan-Jun Xiong, Stella X. Yu, and Da-Hua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition*, 3733–3742.

[43] Hao-Hang Xu, Hong-Kai Xiong, and Guo-Jun Qi. 2021. K-shot contrastive learning of visual features with multiple instance augmentations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 11 (2021), 8694–8700.

[44] Hao-Hang Xu, Xiao-Peng Zhang, Hao Li, Ling-Xi Xie, Wen-Rui Dai, Hong-Kai Xiong, and Qi Tian. 2022. Seed the views: Hierarchical semantic alignment for contrastive representation learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 3 (2022), 3753–3767.

[45] Yan-Wu Xu, Ming-Ming Gong, Jun-Xiang Chen, Tong-Liang Liu, Kun Zhang, and Kayhan Batmanghelich. 2020. Generative-discriminative complementary learning. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, 6526–6533.

[46] Xi Yu Yu, Tong-Liang Liu, Ming-Ming Gong, and Da-Cheng Tao. 2018. Learning with biased complementary labels. In *Proceedings of the 15th European Conference on Computer Vision*, 69–85.

[47] Tong Zhang. 2004. Statistical analysis of some multi-category large margin classification methods. *Journal of Machine Learning Research* 5 (Oct. 2004), 1225–1251.

[48] Zhi-Lu Zhang and Mert R. Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 8792–8802.

[49] Zhi-Hua Zhou. 2018. A brief introduction to weakly supervised learning. *National Science Review* 5, 1 (2018), 44–53.

Appendices

A the Proof of Theorem 2

Theorem 2. *For $\boldsymbol{f}$, there exists a parameter assignment θ to make $\boldsymbol{f}$ a perfect classifier, such that for any $\boldsymbol{x}$, $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$. Then h can assign the ground-truth label for $\boldsymbol{x}$. For any embedding function $\boldsymbol{g}$, fixed Q and $N \to \infty$, it holds that*

$$\bar{L}_{\text{sifted}} \geq L_{\text{ideal}}.$$

Proof. The Dominated Convergence Theorem can be used to prove the above theorem [5]. For the ideal contrastive loss, we have

$$\begin{aligned}
&\lim_{N\to\infty} \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p, \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\\ \boldsymbol{x}_i^-\sim p_{\boldsymbol{x}}^-}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{Q}{N}\sum_{i=1}^N e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p, \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\\ \boldsymbol{x}_i^-\sim p_{\boldsymbol{x}}^-}} \left[\lim_{N\to\infty} -\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{Q}{N}\sum_{i=1}^N e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}}\right] \\
&= \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p, \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + Q\mathbb{E}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} e^{\boldsymbol{q}^T\boldsymbol{k}^-}}\right].
\end{aligned}$$

Since the Dominated Convergence Theorem, the second step of the above equation holds. For the sifted contrastive loss proposed by us, we can obtain

$$\begin{aligned}
&\lim_{N\to\infty} \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\\ \boldsymbol{x}_i^-\sim p}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \sum_{i=1}^N \mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}_i^-))} e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}}\right] \\
&= \lim_{N\to\infty} \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\\ \boldsymbol{x}_i^-\sim p}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N}{N}\sum_{i=1}^N \mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}_i^-))} e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))} e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \log\left(\frac{N}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+} \left[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))} e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right] + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}\right.\right. \\
&\quad \left.\left. \left[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))} e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right] + e^{\boldsymbol{q}^T\boldsymbol{k}^+}\right)\right] \\
&\geq \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \log\left(e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} \left[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))} e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]\right)\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} \left[e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right].
\end{aligned}$$

The second step holds the above equation because of the Dominated Convergence Theorem. The fourth step holds because of using Proposition 1 to decompose the denominator. When $Q = \frac{N(K-1)}{K}$, we have $\bar{L}_{\text{sifted}} \geq L_{\text{ideal}}$. □

B The Proof of Theorem 3

Theorem 3. *There exists a parameter assignment θ which makes $\boldsymbol{f}$ for any $\boldsymbol{x}$ satisfy $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$, rendering it a perfect classifier. Then h can assign the ground-truth label for $\boldsymbol{x}$. For any embedding function $\boldsymbol{g}$ and $N \to \infty$, it holds that*

$$\bar{L}_{\text{sifted}} < L_{\text{std}} - \log\left(1 + \frac{e^{-2}}{K - 1 + K/N}\right).$$

Proof. Similar to the proof of Theorem 2, we have

$$\begin{aligned}
\lim_{N\to\infty}\left[L_{\text{std}} - \bar{L}_{\text{sifted}}\right] &= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[-\log\frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}\right] \\
&\quad - \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[-\log\frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))}e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[-\boldsymbol{q}^T\boldsymbol{k}^+ + \log\left(e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]\right)\right] - \\
&\quad \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[-\boldsymbol{q}^T\boldsymbol{k}^+ + \log\left(e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))}e^{\boldsymbol{q}^T\boldsymbol{k}^-}]\right)\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[\log\frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))}e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[\log\frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}] + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))}e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[\log\frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}] + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[\log\left(1 + \frac{\frac{N}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}\right)\right] \\
&> \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}}\left[\log\left(1 + \frac{\frac{N}{K}e^{-1}}{e + \frac{N(K-1)}{K}e}\right)\right] \\
&= \log\left(1 + \frac{e^{-2}}{K - 1 + K/N}\right).
\end{aligned}$$

Due to the Dominated Convergence Theorem, the first step of the above equation holds. The fourth step of this equation holds because of $\frac{N}{K}\mathbb{E}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+}[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))}e^{\boldsymbol{q}^T\boldsymbol{k}^-}] = 0$, that is this theorem supposes h be a perfect function and $\boldsymbol{x}^-$ is drawn from $p_{\boldsymbol{x}}^+$, so $\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))} = 0$ leads to $\frac{N}{K}\mathbb{E}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+}[\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))}e^{\boldsymbol{q}^T\boldsymbol{k}^-}] = 0$. When $\boldsymbol{x}^-$ is sampled from $p_{\boldsymbol{x}}^-$, we have $\mathbb{I}_{(h(\boldsymbol{x})\neq h(\boldsymbol{x}^-))} = 1$, thus the fifth step holds. The inequality is established since $e^{-1} \leq \mathbb{E}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}], \mathbb{E}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}[e^{\boldsymbol{q}^T\boldsymbol{k}^-}] \leq e$. □

C The Proof of Theorem 4

Theorem 4. *There exists a parameter assignment θ that makes $\boldsymbol{f}$ be a perfect classifier, such that for any $\boldsymbol{x}$, $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$. Then h can assign the ground-truth label for $\boldsymbol{x}$. With any embedding function $\boldsymbol{g}$ and $N \to \infty$, the following inequality holds*

$$L_{\text{ideal}} \leq \bar{L}_{\text{soft}} \leq L_{\text{std}}.$$

Proof. Similar to the proofs of Theorems 2 and 3, we first derive the relationship between the soft contrastive loss $\bar{L}_{\text{soft}}$ and the ideal one. According to the Dominated Convergence Theorem, we can obtain

$$\begin{aligned}
&\lim_{N\to\infty} \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\\ \boldsymbol{x}_i^-\sim p}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \sum_{i=1}^N \left(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}_i^-)\right) e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}\left[\left(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)\right) e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+}\left[\left(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)\right) e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right] + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}\left[\left(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)\right) e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right] \\
&\geq \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}\left[\left(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)\right) e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right] \qquad \left(\because \left(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)\right) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \geq 0\right) \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K}\mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-}\left[e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right].
\end{aligned}$$

Since Theorem 4 defines that $\boldsymbol{f}$ is a perfect classifier and h can assign the ground-truth label of $\boldsymbol{x}$, the ground-truth label of $\boldsymbol{x}^-$ sampled from $p_{\boldsymbol{x}}^-$ differs from that of $\boldsymbol{x}$. So, we have $f_{h(\boldsymbol{x})}(\boldsymbol{x}^-) = 0$ to make the last equation of above hold. When $Q = \frac{N(K-1)}{K}$, we have $\bar{L}_{\text{soft}} \geq L_{\text{ideal}}$. In addition, we establish the upper bound of $\bar{L}_{\text{soft}}$ based on L_{std}, that is

$$\begin{aligned}
\lim_{N\to\infty}\left[L_{\text{std}} - \bar{L}_{\text{soft}}\right] = &\mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}\left[e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right] \\
&- \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p}\left[\left(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)\right) e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]}\right]
\end{aligned}$$

$$
\begin{aligned}
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\boldsymbol{q}^T\boldsymbol{k}^+ + \log\left(e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}] \right) \right] - \\
&\mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\boldsymbol{q}^T\boldsymbol{k}^+ + \log\left(e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right] \right) \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[(1 - f_{h(\boldsymbol{x})}(\boldsymbol{x}^-)) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right] - N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[f_{h(\boldsymbol{x})}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&\geq \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \qquad \left(\because f_{h(\boldsymbol{x})}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \geq 0 \right) \\
&= 0.
\end{aligned}
$$

Based on the above derivation steps, we can obtain $L_{\text{ideal}} \leq \bar{L}_{\text{soft}} \leq L_{\text{std}}$. □

D The Proof of Theorem 5

Theorem 5. *There exists a parameter assignment θ to enable $\boldsymbol{f}$ to be a perfect classifier, such that for any $\boldsymbol{x}$, $\boldsymbol{f}_\theta(\boldsymbol{x}) = p(\boldsymbol{y}|\boldsymbol{x})$. With any embedding function $\boldsymbol{g}$ and $N \to \infty$, it holds that*

$$L_{\text{ideal}} \leq \bar{L}_{\text{wtd}} \leq L_{\text{std}}.$$

Proof. According to the proof of Theorem 4, we can obtain

$$
\begin{aligned}
&\lim_{N\to\infty} \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+,\\ \boldsymbol{x}_i^-\sim p}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \sum_{i=1}^{N} \bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}_i^-) e^{\boldsymbol{q}^T\boldsymbol{k}_i^-}} \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^+} \left[\bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right] + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} \left[\bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&\geq \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} \left[\bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \qquad \left(\because \bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \geq 0 \right)
\end{aligned}
$$

$$= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} \left[\mathbb{I}_{(h(\boldsymbol{x}^-)\in\bar{Y})} \bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} + \mathbb{I}_{(h(\boldsymbol{x}^-)\notin\bar{Y})} \bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right]$$

$$= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} \left[\mathbb{I}_{(h(\boldsymbol{x}^-)\in\bar{Y})} \bar{\boldsymbol{y}}^T \boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right]$$

$$= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + \frac{N(K-1)}{K} \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p_{\boldsymbol{x}}^-} \left[e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right].$$

In the fifth step of the above equation, $\mathbb{I}_{(\cdot)}$ denotes the indicator function. The sixth step holds as $\bar{\boldsymbol{y}}^T\boldsymbol{f}(\boldsymbol{x}^-) = 0$ when $h(\boldsymbol{x}^-) \notin \bar{Y}$. Since $\bar{\boldsymbol{y}}^T\boldsymbol{f}(\boldsymbol{x}^-) = 1$ when $h(\boldsymbol{x}^-) \in \bar{Y}$, such that $\mathbb{I}_{(h(\boldsymbol{x}^-)\in\bar{Y})}\bar{\boldsymbol{y}}^T\boldsymbol{f}(\boldsymbol{x}^-) = 1$ and the last equation holds. When $Q = \frac{N(K-1)}{K}$, we have $\bar{L}_{\text{wtd}} \geq L_{\text{ideal}}$. Subsequently, we explore the upper bound of $\bar{L}_{\text{wtd}}$, that is

$$\begin{aligned}
\lim_{N\to\infty} \left[L_{\text{std}} - \bar{L}_{\text{wtd}} \right] &= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]} \right] \\
&\quad - \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+}}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\bar{\boldsymbol{y}}^T\boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\boldsymbol{q}^T\boldsymbol{k}^+ + \log \left(e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}] \right) \right] - \\
&\quad \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[-\boldsymbol{q}^T\boldsymbol{k}^+ + \log \left(e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\bar{\boldsymbol{y}}^T\boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right] \right) \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\bar{\boldsymbol{y}}^T\boldsymbol{f}(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\sum_{j=1, j\in\bar{Y}}^{K} f_j(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\left(1 - \sum_{j=1, j\notin\bar{Y}}^{K} f_j(\boldsymbol{x}^-) \right) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right] \\
&= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right] - N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[\sum_{j=1, j\notin\bar{Y}}^{K} f_j(\boldsymbol{x}^-) e^{\boldsymbol{q}^T\boldsymbol{k}^-} \right]} \right]
\end{aligned}$$

$$
\begin{aligned}
&> \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}\sim p,\\ \boldsymbol{x}^+\sim p_{\boldsymbol{x}}^+}} \left[\log \frac{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} [e^{\boldsymbol{q}^T\boldsymbol{k}^-}]}{e^{\boldsymbol{q}^T\boldsymbol{k}^+} + N \mathop{\mathbb{E}}_{\boldsymbol{x}^-\sim p} \left[e^{\boldsymbol{q}^T\boldsymbol{k}^-}\right]} \right] \\
&= 0.
\end{aligned}
$$

Combining the derivation steps, we have $L_{\text{ideal}} \leq \tilde{L}_{\text{wtd}} \leq L_{\text{std}}$. □

Received 2 September 2025; revised 16 January 2026; accepted 17 March 2026