ULAREF: A Unified Label Refinement Framework for Learning with Inaccurate Supervision

Congyu Qiao [1 2] **Ning Xu** [1 2] **Yihao Hu** [1 2] **Xin Geng** [1 2]

Abstract

Learning with inaccurate supervision is often encountered in weakly supervised learning, and researchers have invested a considerable amount of time and effort in designing specialized algorithms for different forms of annotations in inaccurate supervision. In fact, different forms of these annotations share the fundamental characteristic that they all still incorporate some portion of correct labeling information. This commonality can serve as a lever, enabling the creation of a cohesive framework designed to tackle the challenges associated with various forms of annotations in learning with inaccurate supervision. In this paper, we propose a unified label refinement framework named ULAREF, i.e., *a Unified LAbel REfinement Framework for learning with inaccurate supervision*, which is capable of leveraging label refinement to handle inaccurate supervision. Specifically, our framework trains the predictive model with refined labels through global detection of reliability and local enhancement using an enhanced model fine-tuned by a proposed consistency loss. Also, we theoretically justify that the enhanced model in local enhancement can achieve higher accuracy than the predictive model on the detected unreliable set under mild assumptions.

1. Introduction

Due to the time-consuming and expensive nature of acquiring large-scale high-quality data, the necessity for weakly supervised learning has emerged in various real-world scenarios. These scenarios include online queries (Liu et al., 2011), crowdsourcing (Arpit et al., 2017; Ibrahim et al., 2023), ecoinformatics (Tang & Zhang, 2017), multimedia content analysis (Zeng et al., 2013), among others, where handling inaccurate supervision is a challenging problem. Typical learning paradigms related to inaccurate supervision encompass learning with noisy labels (Natarajan et al., 2013; Liu & Tao, 2016), where instances may be annotated with incorrect labels replacing the correct ones, and learning with partial labels (Jin & Ghahramani, 2002; Nguyen & Caruana, 2008), where instances are annotated with a candidate label set that hides the correct label. Under these paradigms, characterized by mistakes or ambiguity, the predictive model tends to overfit to inaccurate annotations, adversely affecting its generalization ability.

To address this challenge, a number of approaches have been proposed. For noisy labels, selection-based strategies (Jiang et al., 2018; Chen et al., 2019; Li et al., 2020) back-propagate the loss for the clean label of an instance obtained by a sample selection algorithm to remove the mistakes in inaccurate annotations. When dealing with partial labels, identification-based strategies (Zhang et al., 2016a; Lv et al., 2020; Zhang et al., 2021) focus on iteratively identify one label from the candidate label set as the correct label to reduce the ambiguity in inaccurate annotations. Besides, to learn from crowdsourcing, correction-based strategies (Zhang et al., 2016b; Ibrahim et al., 2019; Ibrahim & Fu, 2021) correct and integrate labels via estimating the annotators' confusion parameters, which attempts to alleviate both mistakes and ambiguity in inaccurate annotations.

Previously, researchers tend to devote significant time and effort to develop specialized learning algorithms for various forms of annotations under inaccurate supervision. In fact, different forms of these annotations share the fundamental characteristic that they all still incorporate some portion of correct labeling information. This commonality can serve as a lever, enabling the creation of a cohesive framework designed to tackle the challenges associated with various forms of annotations in learning with inaccurate supervision and avoid designing specialized approaches that consume a substantial amount of resources.

In this paper, we propose a unified label refinement framework named ULAREF, i.e., *a Unified LAbel REfinement*

[1]School of Computer Science and Engineering, Southeast University, Nanjing, China [2]Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, China. E-mail: {qiaocy, xning, yhhu, xgeng}@seu.edu.cn. Correspondence to: Ning Xu <xning@seu.edu.cn>, Xin Geng <xgeng@seu.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

Framework for learning with inaccurate supervision, where we optimize our predictive model with the empirical risk estimator using our refined labels. We consider label refinement with the outputs of the predictive model from two aspects. First, we globally detect the reliability of the prediction made by the predictive model. The training instance set is divided into the detected reliable set, the major instances of which the model predicts correctly, and unreliable set, the major instances of which the model predicts incorrectly. Second, we locally enhance the output of the predictive model on the detected unreliable set via an enhanced model. We initialize an enhanced model with the predictive model, which is then fine-tuned on the detected reliable set and the neighbourhood of the set with the proposed consistency loss to perform regularization. Upon fine-tuning, we incorporate the outputs of the enhanced model and predictive model to improve the reliability of the overall label refinement.

Besides, we also provide a theoretical analysis on the process of the local enhancement in our framework. By resorting to a cluster property on instance distribution that an instance shares its correct label with sufficient instances in its neighborhood, we deduce a theorem demonstrating that our enhanced model is able to obtain higher accuracy than the predictive model on the detected unreliable set if it has sufficient consistency on the reliable set, i.e., the consistency between the predictions of the enhanced model and the predictive model on the reliable set, as well as the consistency between the predictions of the enhanced model on the reliable set and the neighborhood of the reliable set. Our contributions are summarized as follows:

- Practically, we propose a unified label refinement framework named ULAREF for inaccurate supervision, which trains the predictive model with refined labels via globally detecting the reliability of the supervision signal provided by the predictive model and locally enhancing supervision signal with a proposed consistency loss.
- Theoretically, we prove that the enhanced model in the process of local enhancement could achieve higher accuracy than the predictive model on the detected unreliable set under mild assumptions, which guarantees the improvement on the overall reliability of label refinement.

Experimental results on two typical paradigms of inaccurate supervision, i.e., noisy label learning (NLL) and partial label learning (PLL), demonstrate the effectiveness of our proposed framework compared with the baselines under the respective settings.

2. Related Work

There are two main aspects of inaccurate supervision, i.e., mistakes and ambiguity. Hence, in this section, we briefly review related works in two typical learning paradigms about inaccurate supervision, i.e., NLL and PLL, under the settings of which we also conduct the corresponding experiments introduced in Section 4.

Noisy labels are caused by mistakes of correct labels during annotating. To reduce its impact, loss-based strategies (Ghosh et al., 2017b; Zhang & Sabuncu, 2018a; Ma et al., 2020; Yao et al., 2020; Englesson & Azizpour, 2021) consider the loss about noisy labels from the aspects of robustness (Ghosh et al., 2017a; Wang et al., 2019; Zhang & Sabuncu, 2018b) and correction (Patrini et al., 2017a; Goldberger & Ben-Reuven, 2017). Architecture-based strategies (Srivastava et al., 2014; Chen & Gupta, 2015; Goldberger & Ben-Reuven, 2016; Bekker & Goldberger, 2016; Han et al., 2018a; Yao et al., 2018) aim to incorporate a noise adaptation layer atop a deep neural network (DNN) for learning the label transition process or design a dedicated architecture to accommodate a broader range of label noise types. Selection-based strategies (Malach & Shalev-Shwartz, 2017; Jiang et al., 2018; Han et al., 2018b; Yu et al., 2019; Li et al., 2020; Zhou et al., 2020) backpropagate the loss associated with the clean label of an instance determined by a sample selection algorithm. Regularization-based strategies (Jenni & Favaro, 2018; Tanno et al., 2019; Hendrycks et al., 2019; Wei et al., 2021b) prevent a DNN from overfitting false-labeled instances by imposing certain training restrictions.

Partial labels increase ambiguity of labeling though avoiding the missing of the correct label. To perform disambiguation, mainstream approaches could be divided into average-based strategies and identification-based strategies. Average-based strategies (Hüllermeier & Beringer, 2006; Cour et al., 2011; Zhang & Yu, 2015) treat each candidate label of the instance equally. Recently, (Lv et al., 2023) propose a theoretically grounded framework for this research line based on an average partial-label loss family. Identification-based strategies (Jin & Ghahramani, 2002; Liu & Dietterich, 2014; Zhang & Yu, 2015; Zhang et al., 2016a), which treat the correct label as a latent variable and aim to identify it from the candidate label set, have been concentrated and attained tremendous improvements. (Lv et al., 2020; Feng et al., 2020) simply use the prediction of predictive model as the label information to put more weights on more possibly correct label. (Zhang et al., 2021; Wang et al., 2022) leverage the inner representation of the predictive model to identify the correct label. (Wu et al., 2022) incorporate the outputs of the predictive model on different data augmentations to form the label distribution. (Xu et al., 2023) set a threshold on the output of the predictive model to eliminate incorrect labels to obtain a purer candidate set.

Researchers before have devoted significant time and effort to develop specialized learning algorithms capable for diverse forms of annotations under inaccurate supervision. In fact, different forms of these annotations share the funda-

mental characteristic that they all still incorporate some portion of correct labeling information. For instance, in NLL, there still exists some samples annotated by correct labels though some samples annotated by incorrectly. In PLL, it is known that correct labels are certain to be in the annotated candidate labels. In this paper, we leverage the commonality as a lever and propose a unified label refinement framework to address these various forms of inaccurate annotations to avoid manually designing specialized approaches, which costs a substantial amount of resources.

3. Proposed Method

First of all, we briefly introduce some necessary notations. Let $\mathcal{X} = \mathbb{R}^q$ be the q-dimensional instance space and $\mathcal{Y} = \{1, 2, ..., c\}$ be the label space with c class labels. The training dataset with inaccurate supervision is denoted by $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{l}_i)|1 \leq i \leq n\}$, where $\boldsymbol{x}_i \in \mathcal{X}$ denotes the i-th q-dimensional instance associated with its correct label $y_{\boldsymbol{x}_i} \in \mathcal{Y}$, and the logical vector $\boldsymbol{l}_i = [l_i^1, l_i^2, \ldots, l_i^c]^\top$ denotes the annotation for $\boldsymbol{x}_i$. The j-th element of $\boldsymbol{l}_i$ represents whether the label j is one of the annotated labels, i.e., $l_i^j = 1$ if j is a label annotated to $\boldsymbol{x}_i$, otherwise $l_i^j = 0$. For noisy labels where mistakes exist, $\forall 1 \leq i \leq n, \sum_{j=1}^{c} l_i^j = 1$, and $\exists 1 \leq i \leq n, l_i^{y_{\boldsymbol{x}_i}} \neq 1$. For partial labels where ambiguity exists, $\forall 1 \leq i \leq n, \sum_{j=1}^{c} l_i^j > 1$, and $l_i^{y_{\boldsymbol{x}_i}} = 1$. The objective of learning with inaccurate supervision is to derive a multi-class predictive model f from $\mathcal{D}$ capable of assigning the correct labels to unobserved instances.

3.1. Overview

In our label refinement framework, we optimize our predictive model using the empirical risk estimator with refined labels. Label refinement in our approach involves two key aspects. Initially, we globally detect prediction reliability, dividing the training instances into a detect reliable set (mainly composed of correctly predicted instances) and unreliable one (mainly composed of incorrectly predicted instances). Subsequently, we locally enhance the predictive model's output on the detected unreliable set using an enhanced model. The enhanced model, initialized with the predictive model, undergoes fine-tuning on the detected reliable set and its unreliable neighborhood, employing the proposed consistency loss for regularization. After fine-tuning, we integrate the outputs of both the enhanced model and predictive model to improve the overall label refinement's reliability.

Moreover, we theoretically analyzes the local enhancement process. Leveraging a cluster property on instance distribution, where an instance shares its correct label with sufficient instances in its neighbours, we establish a theorem demonstrating that the enhanced model can achieve higher accuracy than the predictive model on the detected unreliable set if it exhibits adequate consistency on the reliable set. This consistency includes alignment between predictions of the enhanced model and the predictive model on the reliable set, as well as alignment between predictions of the enhanced model on the reliable set and its neighborhood.

3.2. The ULAREF Framework

3.2.1. Optimization Objective

In this paper, we model the predictive model as a DNN with `softmax` as the final layer, denoted as $f : \mathcal{X} \mapsto \Delta^{c-1}$, where Δ^{c-1} represents the c-dimensional simplex. We optimize our predictive model f by using the following empirical risk estimator $\widehat{R}(f)$ with refined labels:

$$\widehat{R}(f) = -\frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{c} r_i^j \log f_j(\boldsymbol{x}_i), \tag{1}$$

Here, it is essential to note that the refined label of the instance $\boldsymbol{x}_i$, denoted by $\boldsymbol{r}_i = [r_i^1, r_i^2, \ldots, r_i^c] \in [0, 1]^c$, can be considered as a kind of post-processed results of the outputs of predictive model $f$and enhanced model g in the process of local enhancement, which could be computed at epoch t as follows:

$$\boldsymbol{r}_i^{[t]} = \frac{(1-\lambda)\boldsymbol{r}_i^{[t-1]} + \lambda((1-m_i)f(\boldsymbol{x}_i) + m_i g(\boldsymbol{x}_i))}{||(1-\lambda)\boldsymbol{r}_i^{[t-1]} + \lambda((1-m_i)f(\boldsymbol{x}_i) + m_i g(\boldsymbol{x}_i))||_1}, \tag{2}$$

where $\lambda \in [0, 1]$ is a constant weighting the supervision information provided by the original dataset and the models, and $||\cdot||_1$ denotes the L1-norm of a vector used to perform normalization here. We initialize $\boldsymbol{r}_i^{[0]} = \boldsymbol{l}_i$ to start our optimization. Besides, $\boldsymbol{m} = [m_1, m_2, \ldots, m_n]$ is a mask to identify the instances in a detected unreliable set $\mathcal{D}_{\boldsymbol{x}}^{\mathrm{u}}$ during the process of local enhancement below. For each instance $\boldsymbol{x}_i$, the corresponding mask value m_i is decided by:

$$m_i = \begin{cases} \beta, & \text{if } \boldsymbol{x}_i \in \mathcal{D}_{\mathrm{s}}^{\mathrm{u}}, \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where $\beta \in [0, 1]$ is a trade-off for supervision information between the predictive model and enhanced model when the instance $\boldsymbol{x}_i$ exists in the detected unreliable set $\mathcal{D}_{\mathrm{s}}^{\mathrm{u}}$.

3.2.2. Global Detection

Since the label refinement depends on the output of the predictive model, our first operation in the framework ULAREF is to perform global detection on its reliability, which means that we should decide which instance the predictive model may predicts correctly and which instance the predictive model may predicts incorrectly, and then divide our training

Algorithm 1 ULAREF Algorithm

Require: The training dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{l}_i) | 1 \leq i \leq n\}$, iteration T;
for $t = 1, \ldots, T$ **do**
  Initialize the detected reliable set $\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}} = \emptyset$ and unreliable set $\mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}} = \emptyset$;
  if $T_{\mathrm{s}} \leq t \leq T_{\mathrm{e}}$ **then**
    Perform global detection to divide the training instance set $\mathcal{D}_{\boldsymbol{x}}$ into the detected reliable set $\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}$ and unreliable set $\mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}}$ according to Eq. (6) and (8);
    Perform local enhancement via initializing the enhanced model $g = f$ and fine-tuning the enhanced model g with the loss function Eq. (9);
  end if
  for each instance $\boldsymbol{x}_i$ in $\mathcal{D}_{\boldsymbol{x}}$ **do**
    Perform label refinement to generate the refined label $\boldsymbol{r}_i$ for $\boldsymbol{x}_i$ according to Eq. (2);
  end for
  Train the predictive model f on $\mathcal{D}$ with assistence of refined labels according to Eq. (1);
end for
Ensure: The predictive model f.

instance set $\mathcal{D}_{\boldsymbol{x}} = \{\boldsymbol{x} | (\boldsymbol{x}, \boldsymbol{l}) \in \mathcal{D}\}$ into the detected reliable one $\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}$ and detected unreliable one $\mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}}$.

Specifically, ULAREF initiates the global detection from the epoch T_{s} to ensure that the predictive model f possesses some capability to detect the instances with reliable and unreliable predictions from the predictive model f, and ends the global detection at the epoch T_{e}. This means that if $t < T_{\mathrm{s}}$ or $t > T_{\mathrm{e}}$, we keep $\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}} = \mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}} = \emptyset$.

To identify the reliable instance set $\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}$ from the training instance set $\mathcal{D}_{\boldsymbol{x}} = \{\boldsymbol{x} | (\boldsymbol{x}, \boldsymbol{l}) \in \mathcal{D}\}$, we introduce an entropy-based uncertainty score Ξ_i for each training instance $\boldsymbol{x}_i$ as the criterion to determine whether $\boldsymbol{x}_i$ belongs to the reliable set $\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}$ as follows:

$$\Xi_i = -\sum_{j=1}^{c} f(\boldsymbol{x}_i) \log f(\boldsymbol{x}_i). \tag{4}$$

Inspired by (Li et al., 2020), we utilize a two-component one-dimensional Gaussian Mixture Model (GMM) to characterize the distribution of per-instance scores, i.e.,

$$\boldsymbol{\pi} = \mathrm{GMM}([\Xi_1, \ldots, \Xi_n]). \tag{5}$$

Here, $\boldsymbol{\pi} = [\pi_1, \pi_2, \ldots, \pi_n] \in [0, 1]^n$ is a vector which has been normalized using min-max normalization, and π_i can be interpreted as the likelihood that the predictive model predicts correctly on instance $\boldsymbol{x}_i$. We introduce a threshold $\tau(t)$ ($T_{\mathrm{s}} \leq t \leq T_{\mathrm{e}}$) to select instances for constructing the detected reliable set $\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}$, i.e.,

$$\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}} = \{\boldsymbol{x}_i | \pi_i \geq \tau(t), \boldsymbol{x}_i \in \mathcal{D}_{\boldsymbol{x}}\}, \tag{6}$$

where

$$\tau(t) = \tau_{\mathrm{s}} - (t - T_{\mathrm{s}}) \frac{\tau_{\mathrm{s}} - \tau_{\mathrm{e}}}{T_{\mathrm{e}} - T_{\mathrm{s}}} \tag{7}$$

linearly decreases during the training process within a reasonable range ($\min_i \pi_i \leq \tau_{\mathrm{e}} < \tau_{\mathrm{s}} \leq \max_i \pi_i$). Besides, we typically set τ_{e} to $\max(10^{-6}, \min_i \pi_i)$ in our framework. Upon obtaining the detected reliable set, the detected unreliable set is decided by:

$$\mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}} = \mathcal{D}_{\boldsymbol{x}} \setminus \mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}. \tag{8}$$

3.2.3. LOCAL ENHANCEMENT

After finishing the global detection, we consider how to enhance the supervision information provided by the predictive model on the detected unreliable set. We employ an enhanced model g and initialize it by $g = f$ and propose the following loss function to fine-tuning it:

$$\begin{aligned} \mathcal{L} = & \frac{\kappa}{|\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}|} \sum_{\boldsymbol{x}_i \in \mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}} \ell^{\mathrm{c}}(g(\boldsymbol{x}_i), f(\boldsymbol{x}_i)) + \\ & \frac{1-\kappa}{|\mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}|} \sum_{\boldsymbol{x}_i \in \mathcal{D}^{\mathrm{r}}_{\boldsymbol{x}}} \frac{1}{|\hat{\mathcal{B}}(\boldsymbol{x}_i)|} \sum_{\boldsymbol{x}_j \in \hat{\mathcal{B}}(\boldsymbol{x}_i)} \ell^{\mathrm{c}}(g(\boldsymbol{x}_i), g(\boldsymbol{x}_j)), \end{aligned} \tag{9}$$

where $\kappa \in [0, 1]$ acts as a trade-off between the two terms, ℓ^{c} is a consistency loss, and $\hat{\mathcal{B}}(\boldsymbol{x}_i)$ denotes the approximate neighborhood for the given instance $\boldsymbol{x}_i$. In the implementation of our framework, we set the consistency loss $\ell^{\mathrm{c}}(\cdot|\cdot) = \mathrm{KL}(\cdot||\cdot)$, and inspired by (Chidambaram et al., 2022), we select $\hat{\mathcal{B}}(\boldsymbol{x}_i)$ for the instance $\boldsymbol{x}_i$ as follows:

$$\begin{aligned} \hat{\mathcal{B}}(\boldsymbol{x}_i) = \{\boldsymbol{x}' | \boldsymbol{x}' = \varsigma' \cdot \boldsymbol{x}_i + (1 - \varsigma') \cdot \mathrm{NN}(\boldsymbol{x}_i, \mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}}), \\ \varsigma' = \max(\varsigma, 1 - \varsigma), \varsigma \sim \mathrm{Beta}(\sigma, \sigma)\}, \end{aligned} \tag{10}$$

where $\mathrm{NN}(\boldsymbol{x}_i, \mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}}) = \arg\min_{\boldsymbol{x}' \in \mathcal{D}^{\mathrm{u}}_{\boldsymbol{x}}} ||\Phi(\boldsymbol{x}_i) - \Phi(\boldsymbol{x}')||$, σ is the parameter of Beta Distribution with its value set to 1, and $\Phi(\cdot)$ denotes a feature extractor in the predictive model f outputting an embedding for an instance. Eq. (9) could be considered as a variant of manifold regularization (Belkin et al., 2006; Kamnitsas et al., 2018; Luo et al., 2018) with mix-up (Zhang et al., 2017).

Upon completing the fine-tuning, we integrate the supervision information provided by the enhanced model g for training each instance $\boldsymbol{x}_i$ in a moving-average style through the mask vector $\boldsymbol{m}$ in Eq. (1).

Overall, we iteratively perform global detection and local enhancement in our framework, thereby improving the reliability of the label refinement and the performance of the

predictive model. The algorithmic description of our framework ULAREF is presented in Algorithm 1.

3.3. Theoretical Analysis

We conduct theoretical analysis to demonstrate the feasibility of local enhancement in our framework ULAREF, i.e., whether the operation can improve the reliability of label refinement. The analysis builds upon a compact set $\mathcal{S}$ with the associated probability measure P supported on the instance space $\mathcal{X}$. Our assumptions and proofs are motivated by the previous work (Wei et al., 2021a) but extended to a different problem and framework.

Definition 1. *We say that P exhibits the (α, ϵ)-cluster property on a set $\mathcal{S} \subset \mathcal{X}$ if, for any $\mathcal{V} \subset \mathcal{S}$ with $P(\mathcal{V}) > \alpha$, the following condition holds:*

$$P(\mathcal{N}^\star(\mathcal{V}) \setminus \mathcal{S}) - P(\mathcal{V}) \geq \epsilon. \tag{11}$$

Here, $\mathcal{N}^\star(\mathcal{V}) = \{\boldsymbol{x} | \exists \boldsymbol{x}' \in \mathcal{V}, \mathcal{B}(\boldsymbol{x}) \cap \mathcal{B}(\boldsymbol{x}') \neq \emptyset, \text{and } y_{\boldsymbol{x}} = y_{\boldsymbol{x}'}\}$ denotes the neighborhood of $\mathcal{V}$ sharing the same correct label with the corresponding instance in $\mathcal{V}$, where $\mathcal{B}(\boldsymbol{x}') = \{\boldsymbol{x} | \mu(\boldsymbol{x}, \boldsymbol{x}') \leq r\}$ represents a neighborhood of the instance $\boldsymbol{x}'$ given some measure μ of the instance space with the distance r. Definition 1 means that any sufficiently large subset $\mathcal{V}$ of $\mathcal{S}$, i.e., $P(\mathcal{V}) > \alpha$, will have a sufficiently large neighborhood outside of $\mathcal{S}$ sharing the same correct label with the corresponding instance in $\mathcal{V}$, i.e., $P(\mathcal{N}^\star(\mathcal{V}) \setminus \mathcal{S}) - P(\mathcal{V}) \geq \epsilon$.

Let the prediction label of the model f be denoted by $h(\boldsymbol{x}) = \arg\max_{j \in \mathcal{Y}} f_j(\boldsymbol{x})$ for a given instance $\boldsymbol{x}$. Then for the completely unreliable set where the predictive model f always makes incorrect predictions, i.e., $\mathcal{E}_f = \{\boldsymbol{x} | h(\boldsymbol{x}) \neq y_{\boldsymbol{x}}\}$, we make the following assumption:

Assumption 1. *There exist constants $\alpha, \epsilon > 0$, and P exhibits the (α, ϵ)-cluster property on the set $\mathcal{E}_f$.*

Besides, we introduce a mild assumption regarding the prediction consistency of the enhanced model on the completely reliable set $\bar{\mathcal{E}}_f = \{\boldsymbol{x} | h(\boldsymbol{x}) = y_{\boldsymbol{x}}\}$, where the predictive model always makes correct predictions, considering two aspects. Firstly, the prediction of the enhanced model should not deviate significantly from that of the predictive model. Secondly, the prediction for an instance by the enhanced model should closely align with the predictions for the neighborhood of the instance.

Assumption 2. *For the predictions of the enhanced model g on $\bar{\mathcal{E}}_f$, there exists a constant $\varepsilon \geq \epsilon$ such that the consistency with the predictions of the predictive model f and its predictions on the neighborhood is bounded by the correct rate of the predictive model, i.e.,*

$$P(\bar{\mathcal{E}}_f) - P(\mathcal{I}_{f,g} \cap \mathcal{J}_g \cap \bar{\mathcal{E}}_f) \leq \varepsilon. \tag{12}$$

Here, we define $\mathcal{I}_{f,g} = \{\boldsymbol{x} | h(\boldsymbol{x}) = h'(\boldsymbol{x})\}$ as the set where the prediction of the enhanced model g, denoted by $h'(\boldsymbol{x}) = \arg\max_{j \in \mathcal{Y}} g_j(\boldsymbol{x})$, is the same as that of the predictive model f, i.e., $h(\boldsymbol{x})$. Additionally, let $\mathcal{J}_g = \{\boldsymbol{x} | \forall \boldsymbol{x}' \in \mathcal{B}(\boldsymbol{x}), h'(\boldsymbol{x}) = h'(\boldsymbol{x}')\}$ denote the set where the prediction of the enhanced model g on a given instance coincides with its predictions in the neighborhood. Assumption 2 explicitly articulates the aforementioned consistency by quantifying the gap between $P(\mathcal{I}_{f,g} \cap \mathcal{J}_g \cap \bar{\mathcal{E}}_f)$ and $P(\bar{\mathcal{E}}_f)$, assuming the existence of its upper bound ε. In our framework ULAREF, this upper bound manifests in the loss function for fine-tuning the enhanced model. Subsequently, let $\mathcal{E}_g = \{\boldsymbol{x} | h'(\boldsymbol{x}) \neq y_{\boldsymbol{x}}\}$ denote the instance set where the enhanced model g always predicts incorrectly, and we present our main theoretical result related to improvement of label refinement.

Theorem 1. *Under Assumption 1 and Assumption 2, if for the detected unreliable set $\mathcal{U}$, there exists a constant $\delta > \alpha + \frac{3}{2}P(\mathcal{U}) - P(\mathcal{J}_g \cap \mathcal{U}) - \frac{1}{2}P(\mathcal{I}_{f,g} \cap \mathcal{U})$, such that the error of f on the detected unreliable set $\mathcal{U}$ has a lower bound, i.e., $P(\mathcal{E}_f \cap \mathcal{U}) \geq \delta$, we have*

$$P(\mathcal{E}_g \cap \mathcal{U}) < P(\mathcal{E}_f \cap \mathcal{U}). \tag{13}$$

The detailed proof is available in Appendix A.1. Theorem 1 illustrates that our enhanced model g could achieve a higher accuracy on the detected unreliable set $\mathcal{U}$ compared to the predictive model f. The lower bound to the error of f on the detected unreliable set $\mathcal{U}$, i.e., $P(\mathcal{E}_f \cap \mathcal{U}) \geq \delta$ explains why we perform global detection to separate an detected unreliable set from the training instance set instead of random selection. Meanwhile, $P(\mathcal{E}_g \cap \mathcal{U}) < P(\mathcal{E}_f \cap \mathcal{U})$ indicates that the output of the enhanced model g could then be used to perform local enhancement on that of the predictive model f on the detected unreliable set.

4. Experiments

In this section, we empirically validate the effectiveness of our framework on two typical paradigms of inaccurate supervision: noisy label learning (NLL) and partial label learning (PLL), both of which are investigated on the corrupted benchmark datasets and real-world datasets. Our code is available at https://github.com/palm-ml/ularef.

4.1. Noisy Label Learning

4.1.1. DATASETS

For noisy labels, we initiate by randomly selecting a subset of training instances based on the noisy ratio η. Subsequently, for each selected training instance, we replace its correct label with another possible label to create a noisy label. In the symmetric setting, the noisy label can be any label except the correct one, while in the asymmetric setting,

Table 1: Classification accuracy (mean $_{\text{std}}$) of each comparing approach on benchmark datasets with noisy labels

Dataset	CIFAR-10				CIFAR-100			
Type	Symmetric			Asymmetric	Symmetric			Asymmetric
ι	0.2	0.5	0.8	0.4	0.2	0.5	0.8	0.4
CE	85.75 $_{0.58}$●	79.50 $_{0.39}$●	60.89 $_{1.37}$●	83.59 $_{0.70}$●	59.97 $_{0.31}$●	44.81 $_{1.33}$●	19.95 $_{0.80}$●	43.34 $_{0.34}$●
MIXUP	92.80 $_{0.31}$●	86.66 $_{0.44}$●	69.01 $_{3.34}$●	87.48 $_{0.38}$●	67.41 $_{0.78}$●	56.08 $_{0.52}$●	31.26 $_{1.33}$●	49.09 $_{0.77}$●
CO-TEACHING	92.25 $_{0.11}$●	88.81 $_{0.25}$●	57.65 $_{2.03}$●	84.91 $_{1.05}$●	62.14 $_{0.89}$●	45.38 $_{1.26}$●	17.53 $_{1.39}$●	42.93 $_{0.78}$●
DIVIDEMIX	**95.69 $_{0.20}$**○	93.97 $_{0.18}$●	90.82 $_{0.56}$●	<u>92.23 $_{0.16}$</u>	<u>75.57 $_{0.26}$</u>●	71.44 $_{0.29}$●	50.43 $_{1.13}$●	55.57 $_{0.39}$●
ELR	93.17 $_{0.20}$●	91.04 $_{0.31}$●	78.18 $_{1.02}$●	87.50 $_{0.66}$●	72.44 $_{0.25}$●	64.71 $_{0.40}$●	25.20 $_{1.06}$●	<u>72.35 $_{0.53}$</u>●
ELR+	95.13 $_{0.17}$	**94.37 $_{0.11}$**	<u>91.11 $_{0.37}$</u>	91.88 $_{0.45}$●	70.78 $_{0.34}$●	68.29 $_{0.53}$●	53.98 $_{0.38}$●	69.94 $_{1.64}$●
ADACORR	90.42 $_{0.37}$●	88.79 $_{0.24}$●	55.42 $_{0.31}$●	82.45 $_{0.77}$●	65.88 $_{0.67}$●	52.85 $_{0.89}$●	32.43 $_{0.79}$●	49.21 $_{0.76}$●
PROMIX	94.88 $_{0.09}$●	93.23 $_{0.13}$●	83.11 $_{0.43}$●	89.83 $_{0.59}$●	75.43 $_{0.32}$●	<u>71.64 $_{0.47}$</u>●	43.35 $_{0.43}$●	72.13 $_{0.65}$●
SOP	94.02 $_{0.22}$●	93.01 $_{0.22}$●	90.11 $_{0.66}$●	90.46 $_{1.10}$●	73.44 $_{0.42}$●	70.73 $_{0.55}$●	<u>54.03 $_{0.23}$</u>●	69.53 $_{0.68}$●
ULAREF	<u>95.28 $_{0.06}$</u>	<u>94.31 $_{0.13}$</u>	**91.47 $_{0.24}$**	**92.56 $_{0.28}$**	**76.16 $_{0.27}$**	**72.39 $_{0.21}$**	**54.72 $_{0.42}$**	**76.11 $_{0.25}$**

Table 2: Classification accuracy (mean ± std) of comparing algorithms on the real-world datasets with noisy labels.

Dataset	Clothing1M	Webvision
CE	$70.11 \pm 0.18\%$●	$71.96 \pm 0.68\%$●
MIXUP	$71.21 \pm 0.18\%$●	$72.14 \pm 0.24\%$●
CO-TEACHING	$71.86 \pm 0.50\%$●	$69.54 \pm 0.71\%$●
DIVIDEMIX	<u>$74.42 \pm 0.27\%$</u>●	<u>$77.37 \pm 0.27\%$</u>●
ELR	$72.98 \pm 0.19\%$●	$76.65 \pm 0.57\%$●
ELR+	$74.19 \pm 0.10\%$●	$75.91 \pm 0.26\%$●
ADACORR	$73.71 \pm 0.23\%$●	$72.76 \pm 0.63\%$●
PROMIX	$74.18 \pm 0.27\%$●	$76.12 \pm 0.34\%$●
SOP	$71.35 \pm 0.53\%$●	$73.67 \pm 0.90\%$●
ULAREF	$\mathbf{74.78 \pm 0.14}$	$\mathbf{77.80 \pm 0.24}$

only labels similar to the correct label (e.g., dog and cat) are considered potential noisy labels. The noisy ratio ι takes values from the set $\{0.2, 0.5, 0.8\}$ for the symmetric setting and 0.4 for the asymmetric setting, as outlined in (Patrini et al., 2017b). We split 10% from the training dataset as the validation set.

Additionally, we incorporate two large-scale datasets with real-world noisy labels, namely `Clothing1M` (Xiao et al., 2015) and `Webvision` (Li et al., 2017).

4.1.2. BASELINES

The performance of ULAREF for noisy labels is compared against nine baselines:

- CE, which directly uses standard cross-entropy loss to train the predictive model in a batch.
- MIXUP (Zhang et al., 2018), which trains the predictive model using the mixup technique.
- CO-TEACHING (Han et al., 2018b), which trains two models simultaneously, and makes them teach each other given every mini-batch.
- DIVIDEMIX (Li et al., 2020), which treats noisy instances as unlabeled data and utilizes the strategy (Berthelot et al., 2019) with label co-refinement and co-guessing.
- ELR and ELR+(Liu et al., 2020), which capitalize on early learning via regularization preventing memorization of the false labels.
- ADACORR (Zheng et al., 2020), which is theoretically grounded to correct the noisy labels based on the noisy classifier prediction.
- PROMIX (Xiao et al., 2023), which carefully selects, dynamically expands and maximally uses clean sample sets under the devised semi-supervised learning framework.
- SOP (Liu et al., 2022), which models label noise by introducing an additional sparse over-parameterization term and differentiates the underlying corruptions.

We train a PreActResNet-18 for 200 epochs on `CIFAR-10` and `CIFAR-100`, pretrained ResNet-50 for 15 epochs on `Clothing1M`, and Inception-ResNet-V2 for 100 epochs on `Webvision`. The training setup involves stochastic gradient descent (SGD) optimization with momentum set to 0.9, weight decay chosen from $\{10^{-4}, 5 \times 10^{-4}\}$, and batch size selected from $\{32, 64, 128\}$. Learning rates are chosen from the orders of magnitude $\{10^{-3}, 10^{-2}, 10^{-1}\}$. Common data augmentations, including Random Horizontal Flipping, Random Cropping, Cutout (Devries & Taylor, 2017), and Auto Augment (Cubuk et al., 2019), are applied.

4.1.3. EXPERIMENTAL RESULTS

Tables 1 and 2 present the classification accuracy of each comparative approach for noisy labels on benchmark and real-world datasets, respectively. To ensure robustness, we conducted 5 trials with different random seeds, reporting mean and standard deviation. The optimal results are indicated in bold, with the second-best results underlined.

Table 3: Classification accuracy (mean $_{\text{std}}$) of each comparing approach on benchmark datasets with partial labels.

Dataset	CIFAR-10				CIFAR-100			
Type	Uniform			Instance	Uniform			Instance
γ	0.3	0.5	0.7	-	0.03	0.05	0.1	-
IDGP	92.07 $_{0.32}$•	91.04 $_{0.13}$•	88.37 $_{2.50}$•	86.43 $_{0.23}$•	68.19 $_{0.02}$•	67.68 $_{0.34}$•	62.39 $_{0.95}$•	64.38 $_{0.27}$•
PLCR	93.13 $_{0.15}$•	91.97 $_{0.25}$•	89.89 $_{0.28}$•	86.37 $_{0.38}$•	66.49 $_{0.23}$•	65.81 $_{0.64}$•	62.75 $_{0.42}$•	64.12 $_{0.23}$•
PICO	90.70 $_{0.38}$•	88.93 $_{0.27}$•	84.99 $_{0.79}$•	86.16 $_{0.21}$•	62.54 $_{0.43}$•	61.03 $_{0.29}$•	40.99 $_{1.74}$•	62.98 $_{0.38}$•
CAVL	87.38 $_{4.00}$•	72.81 $_{5.57}$•	52.93 $_{9.87}$•	59.67 $_{3.30}$•	57.86 $_{2.43}$•	46.37 $_{3.55}$•	25.83 $_{1.71}$•	52.59 $_{1.01}$•
LWS	82.94 $_{0.66}$•	53.41 $_{2.33}$•	39.70 $_{1.96}$•	37.49 $_{2.82}$•	59.10 $_{0.91}$•	54.31 $_{0.47}$•	40.49 $_{2.88}$•	53.98 $_{0.99}$•
VALEN	89.19 $_{0.49}$•	88.36 $_{0.30}$•	87.29 $_{0.43}$•	86.06 $_{0.33}$•	66.77 $_{0.99}$•	65.97 $_{0.87}$•	65.27 $_{0.18}$•	62.85 $_{0.56}$•
CC	87.41 $_{0.38}$•	85.33 $_{0.58}$•	81.16 $_{0.53}$•	79.96 $_{0.99}$•	65.15 $_{0.61}$•	64.33 $_{0.50}$•	62.47 $_{0.72}$•	62.40 $_{0.84}$•
PRODEN	89.46 $_{0.30}$•	88.98 $_{0.29}$•	87.21 $_{0.42}$•	86.04 $_{0.21}$•	65.76 $_{0.74}$•	65.21 $_{0.60}$•	64.90 $_{0.29}$•	62.56 $_{1.49}$•
ULAREF	**93.72** $_{0.06}$	**92.98** $_{0.29}$	**91.13** $_{0.09}$	**87.16** $_{0.10}$	**69.95** $_{0.26}$	**69.63** $_{0.25}$	**67.93** $_{0.24}$	**66.45** $_{0.29}$

Additionally, •/○ denotes whether ULAREF is statistically superior/inferior (pairwise t-test at a 0.05 significance level) to the compared approach on each dataset. From Tables 1 and 2, ULAREF, we can observe:

- ULAREF always achieves the best performance and significantly outperforms the compared NLL baselines on the benchmark dataset `CIFAR-100` and the real-world datasets `Clothing1M` and `Webvision`.
- Notably, ULAREF exhibits a substantial advantage under asymmetric noise on `CIFAR-100`, and exceeds the performance of the second-best algorithm by 3.76%.
- While not attaining the best outcome for `CIFAR-10` with noisy ratios of 0.2 and 0.5, it consistently secures the second position.
- Overall, as the complexity of label noise grows, our framework achieves a larger advantage.

4.2. Partial Label Learning

4.2.1. DATASETS

For partial labels, we introduce two types of partial labels: instance-independent and instance-dependent. In the instance-independent setting, we employ a flipping probability γ to synthesize candidate labels. Each incorrect label has the same probability of being flipped as candidate labels. The flipping probability γ takes values from the set $\{0.3, 0.5, 0.7\}$ for `CIFAR-10` and $\{0.03, 0.05, 0.1\}$ for `CIFAR-100`. In the instance-dependent setting, the flipping probability of each incorrect label is calculated following the methodology in (Xu et al., 2021). Also, 10% of the training dataset is split for validation.

Moreover, we include five datasets with real-world partial labels across various domains, including `Lost` (Cour et al., 2011), `BirdSong` (Briggs et al., 2012), `MSRCv2` (Liu & Dietterich, 2012), `Soccer Player` (Zeng et al., 2013), and `Yahoo!News` (Guillaumin et al., 2010). The features of these datasets are all extracted. For each real-world PLL dataset, we run the methods with 80%/10%/10% train/validation/test split.

4.2.2. BASELINES

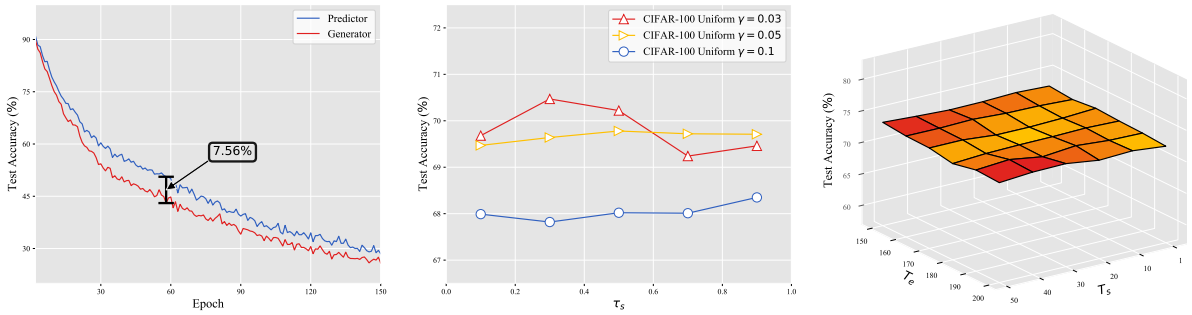
We compare ULAREF with eight methods handling partial labeled data well:

- IDGP (Qiao et al., 2023a), which builds the model upon a decompositional generation process of instance-dependent partial labels.
- PLCR (Wu et al., 2022), which introduces a consistency regularization between feature space and label space.
- PICO (Wang et al., 2022), which adds an additional contrastive loss term to enhance the disambiguation ability.
- CAVL (Zhang et al., 2021), which leverages the class activation value for disambiguation.
- LWS (Wen et al., 2021), which weighs candidate labels and non-candidate labels through a leverage parameter.
- VALEN (Xu et al., 2021), which leverages the variational inference technique to approximate labels distributions for instance-dependent partial labels.
- CC (Feng et al., 2020), which derives a classifier-consistent risk estimator through a transition matrix.
- PRODEN (Lv et al., 2020), which progressively identifies correct labels through the model output.

We implement these approaches using a 32-layer ResNet for benchmark datasets and a linear model for real-world datasets. The optimization involves SGD with momentum set to 0.9, the batch size is set to 256 and the training epochs set to 250. Learning rates are selected from the orders of magnitude $\{10^{-2}, 10^{-3}\}$, and weight decay is chosen from the orders of magnitude $\{10^{-3}, 10^{-3}, 10^{-5}\}$ based on the performance on the validation dataset. The data augmentation techniques are applied to the benchmark datasets in the same manner as that of the noisy label setting. For

Table 4: Classification accuracy (mean ± std) of comparing algorithms on the real-world datasets with partial labels.

	Lost	BirdSong	MSRCv2	Soccer Player	Yahoo!News
IDGP	77.02 ± 0.82%	74.23 ± 0.17%•	50.45 ± 0.47%	**55.99 ± 0.28%**	66.62 ± 0.19%•
CAVL	75.89 ± 0.42%•	73.47 ± 0.13%•	44.73 ± 0.96%•	54.06 ± 0.67%•	65.44 ± 0.23%•
LWS	73.13 ± 0.32%•	51.45 ± 0.26%•	49.85 ± 0.49%•	50.24 ± 0.45%•	48.21 ± 0.29%•
VALEN	76.87 ± 0.86%•	73.39 ± 0.26%•	49.97 ± 0.43%•	55.81 ± 0.10%	66.26 ± 0.13%•
CC	63.54 ± 0.25%•	69.90 ± 0.58%•	41.50 ± 0.44%•	49.07 ± 0.36%•	54.86 ± 0.48%•
PRODEN	76.47 ± 0.25%•	73.44 ± 0.12%•	45.10 ± 0.16%•	54.05 ± 0.15%•	66.14 ± 0.10%•
ULAREF	**77.82 ± 0.89%**	**74.72 ± 0.24%**	**50.56 ± 0.39%**	55.87 ± 0.21%	**67.12 ± 0.14%**



(a) Subset Error (b) Vary τ_s (c) Vary T_s and T_e

Figure 1: Further analysis of ULAREF on CIFAR-100.

real-world PLL datasets, due to the impracticality of implementing data augmentation on extracted features, the data-augmentation-based approaches PLCR and PICO are not compared on PLL real-world datasets. We extend the training epochs to 500 and use a batch size of 100 to maximize algorithms performance. Additionally, due to that the correct label will not be exisits in non-candidate labels, we normalize our refined labels on candidate labels and set the values on non-candidate labels to zero.

4.2.3. EXPERIMENTAL RESULTS

Tables 3 and 4 display the mean and standard deviation of classification accuracy for each comparative approach under five different random seeds on benchmark and real-world datasets with partial labels. We conclude that:

- ULAREF always ranks first and significantly outperforms the compared PLL baselines on all the settings of benchmark datasets CIFAR-10 and CIFAR-100 and the real-world datasets BirdSong and Yahoo!News.
- For the real-world datasets Lost, MSRCv2 and Soccer Player, ULAREF achieves superior or at least comparable performance to other PLL approaches.
- Notably, ULAREF shows more impressive advantage under the settings of CIFAR-100, which is more complex than CIFAR-10.

4.3. Further Analysis

Figure 1(a) visually presents the comparison of error rates between the predictive models and the generator on a fixed unreliable set during the local enhancement process. Notably, in the NLL settings with $\iota = 0.5$ on CIFAR-100, the variation curve of the enhanced model consistently surpasses that of the predictive model. The maximum difference in errors between the enhanced and predictive model reaches up to 7.56%, as indicated in the figure. This observation underscores the effectiveness of the fine-tuning implementation on the enhanced model within our framework ULAREF, which improves the reliability of label refinement.

Furthermore, we conduct sensitivity analyses on key hyperparameters, specifically τ_s for the PLL uniform setting with γ taking values in $0.03, 0.05, 0.1$ on CIFAR-100, and T_s, T_e for the NLL symmetric setting with $\iota = 0.5$ on CIFAR-100. The sensitivity analysis on β and λ can be found in the Appendix A.3. The chosen ranges for these hyperparameters are τ_s from $\{0.1, 0.3, 0.5, 0.7, 0.9\}$, T_s from $\{1, 10, 20, 30, 40, 50\}$, and T_e from $\{150, 160, 170, 180, 190, 200\}$. As depicted in Figures 1(b) and 1(c), the performance of ULAREF remains relatively stable across a range of values for these hyperparameters. This stability signifies robustness, a desirable characteristic for algorithm design.

5. Conclusion

In this paper, we aim at inaccurate supervision and introduce a unified label refinement framework ULAREF, addressing the limitations of previous approaches tailored and confined to their own paradigms. We globally detects the supervision signal provided by the predictive model and locally enhance it with an auxiliary model on the detected unreliable set. Theoretical justification provides a guarantee to improve the reliability of label refinement. Experimental results under typical paradigms of inaccurate supervision confirm the effectiveness of our framework ULAREF.

Acknowledgements

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

Impact Statement

The potential broader impact of our work may lead to increased unemployment among traditional data annotators as the standards for annotation quality decrease. Simultaneously, heightened attention is essential for addressing privacy concerns, given that the utilization of raw data obtained through web scraping can train models with high accuracy.

References

Arpit, D., Jastrzebski, S., Ballas, N., Krueger, D., Bengio, E., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A. C., Bengio, Y., and Lacoste-Julien, S. A closer look at memorization in deep networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pp. 233–242. PMLR, 2017.

Bekker, A. J. and Goldberger, J. Training deep neural-networks based on unreliable labels. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2682–2686. IEEE, 2016.

Belkin, M., Niyogi, P., and Sindhwani, V. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *Journal of machine learning research*, 7(11), 2006.

Berthelot, D., Carlini, N., Goodfellow, I. J., Papernot, N., Oliver, A., and Raffel, C. Mixmatch: A holistic approach to semi-supervised learning. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 5050–5060, 2019.

Briggs, F., Fern, X. Z., and Raich, R. Rank-loss support instance machines for miml instance annotation. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 534–542, 2012.

Chen, P., Liao, B. B., Chen, G., and Zhang, S. Understanding and utilizing deep neural networks trained with noisy labels. In *International Conference on Machine Learning*, pp. 1062–1070. PMLR, 2019.

Chen, X. and Gupta, A. Webly supervised learning of convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 1431–1439, 2015.

Chidambaram, M., Wang, X., Hu, Y., Wu, C., and Ge, R. Towards understanding the data dependency of mixup-style training. *ICLR*, 2022.

Cour, T., Sapp, B., and Taskar, B. Learning from partial labels. *The Journal of Machine Learning Research*, 12: 1501–1536, 2011.

Cubuk, E. D., Zoph, B., Mané, D., Vasudevan, V., and Le, Q. V. Autoaugment: Learning augmentation strategies from data. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 113–123. Computer Vision Foundation / IEEE, 2019.

Devries, T. and Taylor, G. W. Improved regularization of convolutional neural networks with cutout. *CoRR*, abs/1708.04552, 2017.

Englesson, E. and Azizpour, H. Generalized jensen-shannon divergence loss for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34:30284–30297, 2021.

Feng, L., Lv, J., Han, B., Xu, M., Niu, G., Geng, X., An, B., and Sugiyama, M. Provably consistent partial-label learning. *arXiv preprint arXiv:2007.08929*, 2020.

Ghosh, A., Kumar, H., and Sastry, P. S. Robust loss functions under label noise for deep neural networks. In Singh, S. and Markovitch, S. (eds.), *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence,*

February 4-9, 2017, San Francisco, California, USA, pp. 1919–1925. AAAI Press, 2017a.

Ghosh, A., Kumar, H., and Sastry, P. S. Robust loss functions under label noise for deep neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017b.

Goldberger, J. and Ben-Reuven, E. Training deep neural-networks using a noise adaptation layer. In *International Conference on Learning Representations*, 2016.

Goldberger, J. and Ben-Reuven, E. Training deep neural-networks using a noise adaptation layer. In *International Conference on Learning Representations, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Guillaumin, M., Verbeek, J., and Schmid, C. Multiple instance metric learning from automatically labeled bags of faces. In *European conference on computer vision*, pp. 634–647. Springer, 2010.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In Precup, D. and Teh, Y. W. (eds.), *Proceedings of the 34th International Conference on Machine Learning, Sydney, NSW, Australia*, volume 70, pp. 1321–1330, 2017.

Han, B., Yao, J., Niu, G., Zhou, M., Tsang, I., Zhang, Y., and Sugiyama, M. Masking: A new perspective of noisy supervision. *Advances in neural information processing systems*, 31, 2018a.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., and Sugiyama, M. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018b.

Hendrycks, D., Lee, K., and Mazeika, M. Using pre-training can improve model robustness and uncertainty. In *International Conference on Machine Learning*, pp. 2712–2721. PMLR, 2019.

Hüllermeier, E. and Beringer, J. Learning from ambiguously labeled examples. *Intelligent Data Analysis*, 10(5):419–439, 2006.

Ibrahim, S. and Fu, X. Crowdsourcing via annotator co-occurrence imputation and provable symmetric nonnegative matrix factorization. In *International Conference on Machine Learning*, pp. 4544–4554. PMLR, 2021.

Ibrahim, S., Fu, X., Kargas, N., and Huang, K. Crowdsourcing via pairwise co-occurrences: Identifiability and algorithms. *Advances in neural information processing systems*, 32, 2019.

Ibrahim, S., Nguyen, T., and Fu, X. Deep learning from crowdsourced labels: Coupled cross-entropy minimization, identifiability, and regularization. In *11th International Conference on Learning Representations, Kigali, Rwanda*, 2023.

Jenni, S. and Favaro, P. Deep bilevel learning. In *Proceedings of the European Conference on Computer Vision*, pp. 618–633, 2018.

Jiang, L., Zhou, Z., Leung, T., Li, L.-J., and Fei-Fei, L. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pp. 2304–2313. PMLR, 2018.

Jin, R. and Ghahramani, Z. Learning with multiple labels. In *NIPS*, volume 2, pp. 897–904. Citeseer, 2002.

Kamnitsas, K., Castro, D., Le Folgoc, L., Walker, I., Tanno, R., Rueckert, D., Glocker, B., Criminisi, A., and Nori, A. Semi-supervised learning via compact latent space clustering. In *International Conference on Machine Learning*, pp. 2459–2468. PMLR, 2018.

Li, J., Socher, R., and Hoi, S. C. H. Dividemix: Learning with noisy labels as semi-supervised learning. In *8th International Conference on Learning Representations, Addis Ababa, Ethiopia*, 2020.

Li, W., Wang, L., Li, W., Agustsson, E., and Gool, L. V. Webvision database: Visual learning and understanding from web data. *CoRR*, abs/1708.02862, 2017.

Liu, L. and Dietterich, T. Learnability of the superset label learning problem. In *International Conference on Machine Learning*, pp. 1629–1637. PMLR, 2014.

Liu, L. and Dietterich, T. G. A conditional multinomial mixture model for superset label learning. In *Advances in neural information processing systems*, pp. 548–556. Citeseer, 2012.

Liu, S., Niles-Weed, J., Razavian, N., and Fernandez-Granda, C. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

Liu, S., Zhu, Z., Qu, Q., and You, C. Robust training under label noise by over-parameterization. In *International Conference on Machine Learning, 17-23 July 2022, Baltimore, Maryland*, volume 162, pp. 14153–14172. PMLR, 2022.

Liu, T. and Tao, D. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2016.

Liu, W., Jiang, Y., Luo, J., and Chang, S. Noise resistant graph ranking for improved web image search. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pp. 849–856. IEEE Computer Society, 2011.

Luo, Y., Zhu, J., Li, M., Ren, Y., and Zhang, B. Smooth neighbors on teacher graphs for semi-supervised learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8896–8905, 2018.

Lv, J., Xu, M., Feng, L., Niu, G., Geng, X., and Sugiyama, M. Progressive identification of true labels for partial-label learning. In *International Conference on Machine Learning*, pp. 6500–6510. PMLR, 2020.

Lv, J., Liu, B., Feng, L., Xu, N., Xu, M., An, B., Niu, G., Geng, X., and Sugiyama, M. On the robustness of average losses for partial-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–15, 2023.

Ma, X., Huang, H., Wang, Y., Romano, S., Erfani, S., and Bailey, J. Normalized loss functions for deep learning with noisy labels. In *International Conference on Machine Learning*, pp. 6543–6553. PMLR, 2020.

Malach, E. and Shalev-Shwartz, S. Decoupling" when to update" from" how to update". *Advances in neural information processing systems*, 30, 2017.

Natarajan, N., Dhillon, I. S., Ravikumar, P. K., and Tewari, A. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

Nguyen, N. and Caruana, R. Classification with partial labels. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 551–559, 2008.

Patrini, G., Rozza, A., Menon, A. K., Nock, R., and Qu, L. Making deep neural networks robust to label noise: A loss correction approach. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 2233–2241. IEEE Computer Society, 2017a.

Patrini, G., Rozza, A., Menon, A. K., Nock, R., and Qu, L. Making deep neural networks robust to label noise: A loss correction approach. In *IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI*, pp. 2233–2241. IEEE Computer Society, 2017b.

Qiao, C., Xu, N., and Geng, X. Decompositional generation process for instance-dependent partial label learning. In *ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, 2023a.

Qiao, C., Xu, N., Lv, J., Ren, Y., and Geng, X. FREDIS: A fusion framework of refinement and disambiguation for unreliable partial label learning. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning, Honolulu, Hawaii, USA*, 2023b.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Tang, C.-Z. and Zhang, M.-L. Confidence-rated discriminative partial label learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Tanno, R., Saeedi, A., Sankaranarayanan, S., Alexander, D. C., and Silberman, N. Learning from noisy labels by regularized estimation of annotator confusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11244–11253, 2019.

Vishwakarma, H., Lin, H., Sala, F., and Vinayak, R. K. Promises and pitfalls of threshold-based auto-labeling. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36, New Orleans, LA*, 2023.

Wang, H., Xiao, R., Li, Y., Feng, L., Niu, G., Chen, G., and Zhao, J. Pico: Contrastive label disambiguation for partial label learning. *arXiv preprint arXiv:2201.08984*, 2022.

Wang, X., Hua, Y., Kodirov, E., and Robertson, N. M. Imae for noise-robust learning: Mean absolute error does not treat examples equally and gradient magnitude's variance matters. *arXiv preprint arXiv:1903.12141*, 2019.

Wei, C., Shen, K., Chen, Y., and Ma, T. Theoretical analysis of self-training with deep networks on unlabeled data. In *9th International Conference on Learning Representations, Virtual Event, Austria*, 2021a.

Wei, H., Tao, L., Xie, R., and An, B. Open-set label noise can improve robustness against inherent label noise. *Advances in Neural Information Processing Systems*, 34: 7978–7992, 2021b.

Wen, H., Cui, J., Hang, H., Liu, J., Wang, Y., and Lin, Z. Leveraged weighted loss for partial label learning. In *International Conference on Machine Learning*, pp. 11091–11100. PMLR, 2021.

Wu, D.-D., Wang, D.-B., and Zhang, M.-L. Revisiting consistency regularization for deep partial label learning. In *International Conference on Machine Learning*, pp. 24212–24225. PMLR, 2022.

Xiao, R., Dong, Y., Wang, H., Feng, L., Wu, R., Chen, G., and Zhao, J. Promix: Combating label noise via maximizing clean sample utility. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, Macao, China*, pp. 4442–4450, 2023.

Xiao, T., Xia, T., Yang, Y., Huang, C., and Wang, X. Learning from massive noisy labeled data for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pp. 2691–2699. IEEE Computer Society, 2015.

Xu, N., Qiao, C., Geng, X., and Zhang, M.-L. Instance-dependent partial label learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Xu, N., Liu, B., Lv, J., Qiao, C., and Geng, X. Progressive purification for instance-dependent partial label learning. In *International Conference on Machine Learning*, pp. 38551–38565. PMLR, 2023.

Yao, J., Wang, J., Tsang, I. W., Zhang, Y., Sun, J., Zhang, C., and Zhang, R. Deep learning from noisy image labels with quality embedding. *IEEE Transactions on Image Processing*, 28(4):1909–1922, 2018.

Yao, Y., Liu, T., Han, B., Gong, M., Deng, J., Niu, G., and Sugiyama, M. Dual t: Reducing estimation error for transition matrix in label-noise learning. *Advances in neural information processing systems*, 33:7260–7271, 2020.

Yu, X., Han, B., Yao, J., Niu, G., Tsang, I., and Sugiyama, M. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pp. 7164–7173. PMLR, 2019.

Yuan, H., Shi, Y., Xu, N., Yang, X., Geng, X., and Rui, Y. Learning from biased soft labels. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *Advances in Neural Information Processing Systems 36, New Orleans, LA, USA*, 2023.

Zeng, Z., Xiao, S., Jia, K., Chan, T.-H., Gao, S., Xu, D., and Ma, Y. Learning by associating ambiguously labeled images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 708–715, 2013.

Zhang, F., Feng, L., Han, B., Liu, T., Niu, G., Qin, T., and Sugiyama, M. Exploiting class activation value for partial-label learning. In *International Conference on Learning Representations*, 2021.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Zhang, H., Cissé, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. In *6th International Conference on Learning Representations, Vancouver, BC, Canada*, 2018.

Zhang, M.-L. and Yu, F. Solving the partial label learning problem: An instance-based approach. In *Twenty-fourth international joint conference on artificial intelligence*, 2015.

Zhang, M.-L., Zhou, B.-B., and Liu, X.-Y. Partial label learning via feature-aware disambiguation. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1335–1344, 2016a.

Zhang, Y., Chen, X., Zhou, D., and Jordan, M. I. Spectral methods meet em: A provably optimal algorithm for crowdsourcing. *The Journal of Machine Learning Research*, 17(1):3537–3580, 2016b.

Zhang, Z. and Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31, 2018a.

Zhang, Z. and Sabuncu, M. R. Generalized cross entropy loss for training deep neural networks with noisy labels. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 8792–8802, 2018b.

Zheng, S., Wu, P., Goswami, A., Goswami, M., Metaxas, D., and Chen, C. Error-bounded correction of noisy labels. In *International Conference on Machine Learning*, pp. 11447–11457. PMLR, 2020.

Zhou, T., Wang, S., and Bilmes, J. Robust curriculum learning: From clean label detection to noisy label self-correction. In *International Conference on Learning Representations*, 2020.

A. Appendix

A.1. Proofs of Theorem 1

Table 5: Basic notations and their descriptions.

Notation	Meaning
$\mathcal{X}$	the instance space
$\mathcal{Y}$	the label space
$\mathcal{D}$	training dataset
$\boldsymbol{x}_i$	the i-th instance
$y_{\boldsymbol{x}_i}$	the correct label of the instance $\boldsymbol{x}_i$
$\boldsymbol{l}_i$	a logical vector, denoting the annotation for the instance $\boldsymbol{x}_i$
$\widehat{R}(\cdot)$	an empirical risk estimator
$f(\cdot)$	the predictive model
$\phi(\cdot)$	a feature extractor in the predictive model $f(\cdot)$
$g(\cdot)$	the enhanced model
$\boldsymbol{r}_i$	the refined label for the instance $\boldsymbol{x}_i$
$\mathcal{D}_{\boldsymbol{x}}$	the instance set
$\mathcal{D}_{\boldsymbol{x}}^r$	the detected reliable set during global detection
$\mathcal{D}_{\boldsymbol{x}}^u$	the detected unreliable set during global detection
$\boldsymbol{m}$	a vector, the i-th element of which denotes whether the instance $\boldsymbol{x}_i$ belongs to $\mathcal{D}_{\boldsymbol{x}}^u$
$\boldsymbol{\Xi}_i$	the uncertainty score vector for the instance $\boldsymbol{x}_i$
$\boldsymbol{\pi}$	a vector output by a Gaussian Mixture Model, the i-th element of which can be considered as a estimation of the probability that the predictive model f predicts correctly on the instance $\boldsymbol{x}_i$
τ	a threshold to select instances according to $\boldsymbol{\pi}$ for constructing the detected reliable set
$\mathcal{L}$	the fine-tining loss for the enhanced model g
$\ell_c(\cdot,\cdot)$	a consistency loss, which we instantiate with KL divergence
$\widehat{\mathcal{B}}(\boldsymbol{x}_i)$	the approximate neighborhood for the instance $\boldsymbol{x}_i$
$\mathrm{NN}(\cdot,\cdot)$	nearest-neighbour function
$\mathrm{Beta}(\cdot,\cdot)$	Beta distribution
$\mathcal{N}^\star(\mathcal{V})$	a neighborhood of the set $\mathcal{V}$
$\mathcal{B}(\boldsymbol{x})$	a neighborhood of the instance $\boldsymbol{x}$
$h(\boldsymbol{x})$	the label predicted by the predictive model f on the instance $\boldsymbol{x}$
$h'(\boldsymbol{x})$	the label predicted by the enhanced model g on the instance $\boldsymbol{x}$
$\mu(\cdot,\cdot)$	a function measuring two instance, e.g., Euclidean distance
$\mathcal{E}_f$	the completely unreliable set where the predictive model f always makes incorrect predictions
$\bar{\mathcal{E}}_f$	the completely reliable set where the predictive model f always makes incorrect predictions, where $\bar{\cdot}$ is the inverse operation of the set
$\mathcal{I}_{f,g}$	the instance set where the prediction of the enhanced model g is the same with that of the predictive model f
$\mathcal{J}_g$	the instance set where the prediction of the enhanced model g on a given instance $\boldsymbol{x}$ coincides with the predictions in its neighborhood $\mathcal{B}(\boldsymbol{x})$
$\mathcal{E}_g$	the instance set where the enhanced model g always predicts incorrectly

We could begin with decomposing $P(\mathcal{E}_g \cap \mathcal{U})$ into $P(\mathcal{E}_g \cap \mathcal{J}_g \cap \mathcal{U})$ and $P(\bar{\mathcal{J}}_g \cap \mathcal{U})$ according to the principle of inclusion-exclusion. And then $P(\mathcal{E}_g \cap \mathcal{J}_g \cap \mathcal{U})$ could further be decomposed into two parts $P(\mathcal{E}_1 \cup \mathcal{E}_2)$ and $P(\mathcal{E}_3)$ as we defined later, since $\mathcal{E}_1 \cup \mathcal{E}_2 \cup \mathcal{E}_3 = \mathcal{E}_g \cap \mathcal{J}_g \cap \mathcal{U}$. We bound $P(\mathcal{E}_1 \cup \mathcal{E}_2)$ in Lemma 1, i.e., $P(\mathcal{E}_1 \cup \mathcal{E}_2) \leq \alpha$, and $\mathcal{E}_3$ in Lemma 2, i.e., $P(\mathcal{E}_3) \leq \alpha + P(\bar{\mathcal{J}}_g \cap \mathcal{U}) + P(\bar{\mathcal{I}}_{f,g}) - P(\mathcal{E}_f \cap \mathcal{U})$, according to Assumption 1 and 2. Finally, we combine the above conditions and obtain $P(\mathcal{E}_g \cap \mathcal{U}) < P(\mathcal{E}_f \cap \mathcal{U})$.

Towards proving Theorem 1, we consider three disjoint subsets of $\mathcal{E}_g \cap \mathcal{J}_g \cap \mathcal{U}$:

- $\mathcal{E}_1 = \{\boldsymbol{x} : g(\boldsymbol{x}) = f(\boldsymbol{x}), f(\boldsymbol{x}) \neq y_{\boldsymbol{x}}, \text{and } \boldsymbol{x} \in \mathcal{J}_g \cap \mathcal{U}\}$;
- $\mathcal{E}_2 = \{\boldsymbol{x} : g(\boldsymbol{x}) \neq f(\boldsymbol{x}), f(\boldsymbol{x}) \neq y_{\boldsymbol{x}}, g(\boldsymbol{x}) \neq y_{\boldsymbol{x}}, \text{and } \boldsymbol{x} \in \mathcal{J}_g \cap \mathcal{U}\}\}$;

- $\mathcal{E}_3 = \{\boldsymbol{x} : g(\boldsymbol{x}) \neq f(\boldsymbol{x}), f(\boldsymbol{x}) = y_{\boldsymbol{x}}, \text{and } \boldsymbol{x} \in \mathcal{J}_g \cap \mathcal{U}\}$.

We could first deduce the following lemma about $\mathcal{E}_1 \cap \mathcal{E}_2$:

Lemma 1. *Under Assumption 1 and Assumption 2, we have $P(\mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U}) \leq \alpha$. As a result, since it holds that $\mathcal{E}_1 \cup \mathcal{E}_2 \subset \mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U}$, it immediately follows that $P(\mathcal{E}_1 \cup \mathcal{E}_2) \leq \alpha$.*

We divide the proof of Lemma 1 into the proof of Claim 1 and Claim 2.

Claim 1. *Under Assumption 1 and 2, define $\mathcal{O} = \mathcal{N}^\star(\mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U}) \setminus \mathcal{E}_f$. For any $\boldsymbol{x} \in \mathcal{O} \cap \mathcal{J}_g$, it holds that $f(\boldsymbol{x}) \neq g(\boldsymbol{x})$ and $g(\boldsymbol{x}) \neq y_{\boldsymbol{x}}$.*

Proof. For any $\boldsymbol{x} \in \mathcal{O} \subset \mathcal{N}^\star(\mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U})$, there exists $\boldsymbol{x}' \in \mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U}$ such that $\mathcal{B}(\boldsymbol{x}) \cap \mathcal{B}(\boldsymbol{x}') \neq \emptyset$ and $y_{\boldsymbol{x}} = y_{\boldsymbol{x}'}$ by the definition of $\mathcal{N}^\star(\cdot)$. As $\boldsymbol{x}' \in \mathcal{J}_g$, by the definition of $\mathcal{J}_g$, we also must have $g(\boldsymbol{x}) = g(\boldsymbol{x}')$. Furthermore, as $\boldsymbol{x}' \in \mathcal{E}_g$, $g(\boldsymbol{x}') \neq y_{\boldsymbol{x}'}$. Since $y_{\boldsymbol{x}} = y_{\boldsymbol{x}'}$, it follows that $g(\boldsymbol{x}) \neq y_{\boldsymbol{x}}$.

As $\mathcal{O} \cap \mathcal{E}_f = \emptyset$ by the definition of $\mathcal{O}$, f must match the ground-truth predictive model on $\mathcal{O}$, so $f(\boldsymbol{x}) = y_{\boldsymbol{x}}$. It follows that $f(\boldsymbol{x}) \neq g(\boldsymbol{x})$, as desired.

Claim 2. *Under Assumption 1 and 2, define $\mathcal{O} = (\mathcal{N}^\star(\mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U}) \setminus \mathcal{E}_f$. If $P(\mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U}) > \alpha$, then*

$$P(\mathcal{O} \cap \mathcal{J}_g) > P(\mathcal{E}_g) + P(\mathcal{J}_g) + \epsilon - 1 - P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{E}}_g \cap \mathcal{U}) - P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{U}})$$

Proof. Define $\mathcal{Q} = \mathcal{J}_g \cap \mathcal{E}_g \cap \mathcal{E}_f \cap \mathcal{U}$. By the assumption that $\mathcal{E}_f$ satisfies (α, ϵ)-cluster property, if $P(\mathcal{Q}) > \alpha$ holds, it follows that $P(\mathcal{O}) > P(\mathcal{Q}) + \epsilon$. Furthermore, we have $\mathcal{O} \setminus \mathcal{J}_g \subset \overline{\mathcal{J}_g \cup \mathcal{E}_f}$ by the definition of $\mathcal{O}$ as $\mathcal{O} \cap \mathcal{E}_f = \emptyset$, and so $P(\mathcal{O} \setminus \mathcal{J}_g) \leq 1 - P(\mathcal{J}_g \cup \mathcal{E}_f)$. Thus we obtain

$$P(\mathcal{O} \cap \mathcal{J}_g) = P(\mathcal{O}) - P(\mathcal{O} \setminus \mathcal{J}_g) > P(\mathcal{Q}) + \epsilon - 1 + P(\mathcal{J}_g \cup \mathcal{E}_f).$$

Now we use the principle of inclusion-exclusion to compute

$$P(\mathcal{J}_g \cup \mathcal{E}_f) = P(\mathcal{E}_f) + P(\mathcal{J}_g) - P(\mathcal{J}_g \cap \mathcal{E}_f)$$

Plugging into the previous, we obtain

$$\begin{aligned}
&P(\mathcal{O} \cap \mathcal{J}_g) > P(\mathcal{E}_f) + P(\mathcal{J}_g) - P(\mathcal{J}_g \cap \mathcal{E}_f) + P(\mathcal{Q}) + \epsilon - 1 \\
&= P(\mathcal{E}_f) + P(\mathcal{J}_g) + \epsilon - 1 - P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{E}_g \cap \mathcal{U}}) \\
&= P(\mathcal{E}_f) + P(\mathcal{J}_g) + \epsilon - 1 - P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{E}}_g) - P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{U}}) \\
&+ P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{E}}_g \cap \overline{\mathcal{U}}) \\
&= P(\mathcal{E}_f) + P(\mathcal{J}_g) + \epsilon - 1 - P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{E}}_g \cap \mathcal{U}) - P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{U}})
\end{aligned}$$

To complete the proof of Lemma 1, we first compose $\mathcal{J}_g$ into four disjoint sets:

- $\mathcal{J}_1 = \mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{U}}$;
- $\mathcal{J}_2 = \{\boldsymbol{x} : g(\boldsymbol{x}) = f(\boldsymbol{x})\} \cap \overline{\mathcal{E}}_f \cap \mathcal{J}_g$;
- $\mathcal{J}_3 = \{\boldsymbol{x} : g(\boldsymbol{x}) \neq f(\boldsymbol{x})\} \cap \overline{\mathcal{E}}_f \cap \mathcal{J}_g$;
- $\mathcal{J}_4 = \mathcal{J}_g \cap \mathcal{E}_f \cap \mathcal{U}$.

First, by Claim 1 and the definition of $\mathcal{O}$, we have $\forall \boldsymbol{x} \in \mathcal{O} \cap \mathcal{J}_g$, $g(\boldsymbol{x}) \neq f(\boldsymbol{x})$ and $\boldsymbol{x} \in \overline{\mathcal{E}}_f$. Thus, it follows that $\mathcal{O} \cap \mathcal{J}_g \subset \mathcal{J}_3$.

Next, we claim that $\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{E}}_g \cap \mathcal{U} \subset \mathcal{J}_4$. Assume for the sake of contradiction that $P(\mathcal{Q}) > \alpha$. Now we have

$$\begin{aligned}
&P(\mathcal{J}_g \cap \overline{\mathcal{U}}) = P(\mathcal{J}_1) + P(\mathcal{J}_2) + P(\mathcal{J}_3) + P(\mathcal{J}_4) \\
&\geq P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{U}}) + P(\mathcal{J}_g \cap \mathcal{E}_f \cap \overline{\mathcal{E}}_g \cap \mathcal{U}) + P(\mathcal{J}_2) + P(\mathcal{O} \cap \mathcal{J}_g) \\
&> P(\mathcal{J}_2) + P(\mathcal{E}_f) + P(\mathcal{J}_g) + \epsilon - 1
\end{aligned}$$

However, we also have

$$\begin{aligned} P(\mathcal{J}_2) &= 1 - \mathbb{E}_P[\mathbb{I}[g(\boldsymbol{x}) \neq f(\boldsymbol{x}) \vee \boldsymbol{x} \notin \mathcal{J}_g \vee f(\boldsymbol{x}) \neq y_{\boldsymbol{x}}] \\ &\geq 1 - P(\mathcal{E}_f) - \epsilon \end{aligned}$$

Plugging this in gives us $P(\mathcal{J}_1) + P(\mathcal{J}_2) + P(\mathcal{J}_3) + P(\mathcal{J}_4) > P(\mathcal{J}_g)$, a contradiction. Thus $P(\mathcal{Q}) \leq \alpha$, as desired.

According to Claim 2, due to that $\mathcal{E}_1 \cup \mathcal{E}_2 \subset \mathcal{Q}$, $P(\mathcal{E}_1 \cup \mathcal{E}_2) \leq (\mathcal{Q}) \leq \alpha$. The proof of Lemma 1 has been completed.

Next, we could deduce the following lemma about $\mathcal{E}_3$:

Lemma 2. *Under Assumption 1 and Assumption 2, the following bound holds:*

$$P(\mathcal{E}_3) \leq \alpha + P(\bar{\mathcal{J}}_g \cap \mathcal{U}) + P(\bar{\mathcal{I}}_{f,g} \cap \mathcal{U}) - P(\mathcal{E}_f \cap \mathcal{U})$$

Proof. The proof will follow from basic manipulation. First, we note that

$$\begin{aligned} &\mathcal{E}_3 \cup \{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x}), \text{and } \boldsymbol{x} \in \mathcal{J}_g) \cap \mathcal{U}\} \\ =&(\{\boldsymbol{x} : g(\boldsymbol{x}) \neq f(\boldsymbol{x}), f(\boldsymbol{x}) = y_{\boldsymbol{x}}\} \cap \mathcal{J}_g) \cap \mathcal{U}) \cup (\{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x})\} \cap \mathcal{J}_g) \cap \mathcal{U}) \\ =&(\{\boldsymbol{x} : g(\boldsymbol{x}) \neq f(\boldsymbol{x}), f(\boldsymbol{x}) = y_{\boldsymbol{x}}\} \cup \{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x})\}) \cap \mathcal{J}_g) \cap \mathcal{U} \\ =&(\{\boldsymbol{x} : f(\boldsymbol{x}) = y_{\boldsymbol{x}}\} \cup \{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x})\}) \cap \mathcal{J}_g) \cap \mathcal{U} \\ =&(\{\boldsymbol{x} : f(\boldsymbol{x}) = y_{\boldsymbol{x}}\} \cup \{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x}), f(\boldsymbol{x}) \neq y_{\boldsymbol{x}}\}) \cap \mathcal{J}_g) \cap \mathcal{U} \\ =&\mathcal{E}_1 \cup \{\boldsymbol{x} : f(\boldsymbol{x}) = y_{\boldsymbol{x}}, \text{and } \boldsymbol{x} \in \mathcal{J}_g) \cap \mathcal{U}\} \end{aligned}$$

It follows that

$$\begin{aligned} P(\mathcal{E}_3) + &P(\{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x}), \text{and } \boldsymbol{x} \in \mathcal{J}_g) \cap \mathcal{U}\}) = \\ &P(\mathcal{E}_1) + P(\{\boldsymbol{x} : f(\boldsymbol{x}) = y_{\boldsymbol{x}}, \text{and } \boldsymbol{x} \in \mathcal{J}_g) \cap \mathcal{U}\}) \end{aligned}$$

Thus, we can obtain

$$\begin{aligned} &P(\mathcal{E}_3) = P(\mathcal{E}_1) + P(\{\boldsymbol{x} : f(\boldsymbol{x}) = y_{\boldsymbol{x}}\} \cap \mathcal{J}_g) \cap \mathcal{U}) - P(\{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x})\} \cap \mathcal{J}_g) \cap \mathcal{U}) \\ &\leq P(\mathcal{E}_1) + P(\{\boldsymbol{x} : f(\boldsymbol{x}) = y_{\boldsymbol{x}}\} \cap \mathcal{U}) - P(\{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x})\} \cap \mathcal{J}_g) \cap \mathcal{U}) \\ &\leq P(\mathcal{E}_1) + P(\{\boldsymbol{x} : f(\boldsymbol{x}) = y_{\boldsymbol{x}}\} \cap \mathcal{U}) - P(\{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x})\} \cap \mathcal{U}) \\ &+ P(\{\boldsymbol{x} : f(\boldsymbol{x}) = g(\boldsymbol{x})\} \cap \mathcal{U} \cap \overline{\mathcal{J}}_g) \\ &\leq P(\mathcal{E}_1) + P(\mathcal{U}) - P(\{\boldsymbol{x} : f(\boldsymbol{x}) \neq y_{\boldsymbol{x}}\} \cap \mathcal{U}) - (P(\mathcal{U}) - P(\{\boldsymbol{x} : f(\boldsymbol{x}) \neq g(\boldsymbol{x})\} \cap \mathcal{U})) \\ &+ P(\mathcal{U} \cap \overline{\mathcal{J}}_g) \\ &= P(\mathcal{E}_1) + P(\bar{\mathcal{J}}_g \cap \mathcal{U}) + P(\bar{\mathcal{I}}_{f,g} \cap \mathcal{U}) - P(\mathcal{E}_f \cap \mathcal{U}) \\ &\leq \alpha + P(\bar{\mathcal{J}}_g \cap \mathcal{U}) + P(\bar{\mathcal{I}}_{f,g} \cap \mathcal{U}) - P(\mathcal{E}_f \cap \mathcal{U}) \end{aligned}$$

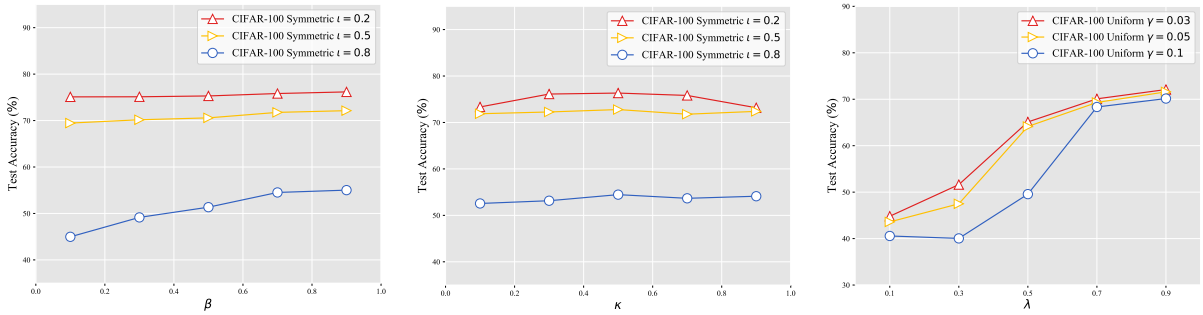
Finally, according to Lemma 1 and 2, we could obtain:

$$\begin{aligned} &P(\mathcal{E}_g \cap \mathcal{U}) \leq P(\mathcal{E}_g \cap \mathcal{J}_g \cap \mathcal{U}) + P(\bar{\mathcal{J}}_g \cap \mathcal{U}) \\ &\leq P(\mathcal{E}_1) + P(\mathcal{E}_2) + P(\mathcal{E}_3) + P(\bar{\mathcal{J}}_g \cap \mathcal{U}) \\ &\leq 2(\alpha + P(\bar{\mathcal{J}}_g \cap \mathcal{U})) + P(\bar{\mathcal{I}}_{f,g} \cap \mathcal{U}) - P(\mathcal{E}_f \cap \mathcal{U}) \\ &< P(\mathcal{E}_f \cap \mathcal{U}) \end{aligned}$$

Here, we finish our proof of Theorem 1.

A.2. Extended experiments

Sensitivity Analysis. We conduct the sensitivity analysis about the hyper-parameters β and κ on `CIFAR-100` under the NLL symmetric setting with ι taking values in $\{0.2, 0.5, 0.8\}$ and λ on `CIFAR-100` under the PLL uniform setting with γ taking values in $\{0.03, 0.05, 0.1\}$. We vary β, κ and λ from $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. As shown in Figure 2(a), 2(b) and



(a) `Vary` β (b) `Vary` κ (c) `Vary` λ

Figure 2: Sensitivity analysis of ULAREF on `CIFAR-100`.

Table 6: Classification accuracy (mean $_{\text{std}}$) of comparing methods on `CIFAR-10` with both noisy and partial labels.

IDGP	PLCR	PICO	CAVL	LWS	VALEN	CC	PRODEN	ULAREF
40.26 $_{2.34}$	46.13 $_{0.92}$	61.39 $_{2.64}$	14.31 $_{2.08}$	12.49 $_{2.34}$	58.77 $_{1.52}$	43.57 $_{2.36}$	58.77 $_{1.52}$	**65.25** $_{\mathbf{1.65}}$

2(c), we can find that the performance of the proposed ULAREF is relatively stable for varying β and κ, which indicates its robustness, and with respect to λ, a recommended range of values is around 0.9.

Model Calibration. We investigate the effect of model calibration by introducing a temperature-scaling parameter, denoted as T, into the softmax function within our global detection procedure, a technique recommended for calibrating model predictions in classical literature (Guo et al., 2017). Figure 3 presents the performance of ULAREF across varying values of T. The figure illustrates that calibration indeed influences ULAREF's performance, with poorly calibrated models (e.g., $T = 0.7$ or 10) exhibiting decreased performance.

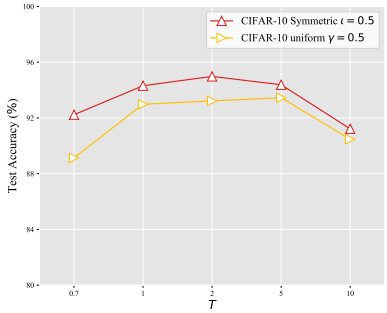


Figure 3: The performance of ULAREF under different temperature T for model calibration on `CIFAR-10`.

Optimally Working. Inspired by (Lv et al., 2023; Qiao et al., 2023b; Yuan et al., 2023), we manually corrupt `CIFAR-10` to obtain a dataset with both partial labels and noisy labels, where NLL baselines cannot handle this setting, and the performance of PLL baselines drop a lot due to label noise. This highlights the distinct advantage of our framework. We control the noisy rate ι at 0.3 and the partial rate γ at 0.5. As depicted in Table 6, ULAREF demonstrates a substantial superiority over other baselines. We intend to include these experimental findings in our revised version.

Ablation Study. We investigate another measure to determine whether the instance $\boldsymbol{x}_i$ belongs to the reliable set $\mathcal{D}^r_{\boldsymbol{x}_i}$:

$$\Xi_i = \max_j f_j(\boldsymbol{x}_i) - \max_{k \neq \arg\max_j f_j(\boldsymbol{x}_i)} f_k(\boldsymbol{x}_i). \tag{14}$$

And we create a variant ULAREF-M and report its performance in Table 7, from which we could observe that ULAREF-M could still achieve comparable results.

Table 7: Ablation study on the choice of the uncertainty score in ULAREF.

Dataset	CIFAR-10				CIFAR-100			
Type	Symmetric			Asymmetric	Symmetric			Asymmetric
ι	0.2	0.5	0.8	0.4	0.2	0.5	0.8	0.4
ULAREF-M	94.56 $_{0.12}$	**94.53** $_{\mathbf{0.22}}$	91.13 $_{0.28}$	92.03 $_{0.45}$	75.87 $_{0.37}$	**72.45** $_{\mathbf{0.25}}$	53.22 $_{0.57}$	75.34 $_{0.31}$
ULAREF	**95.28** $_{\mathbf{0.06}}$	94.31 $_{0.13}$	**91.47** $_{\mathbf{0.24}}$	**92.56** $_{\mathbf{0.28}}$	**76.16** $_{\mathbf{0.27}}$	72.39 $_{0.21}$	**54.72** $_{\mathbf{0.42}}$	**76.11** $_{\mathbf{0.25}}$

A.3. Technical differences from self-training and auto-labeling

Our framework distinguishes itself from self-training and auto-labeling methodologies. We contribute to the community by introducing a novel theorem, demonstrating the potential to obtain greater accuracy with our enhanced model on certain unreliable instances compared to the predictive model, achieved through maintaining local consistency in the enhanced model. Supported by the theorem, we design two procedures in our framework, i.e., global detection and local enhancement. The former detects reliable and unreliable instances, and the latter fine-tunes the enhanced model with the proposed consistency loss. In this way, we get more reliable supervision information locally from the enhanced model. However, the previous works in self-training and auto-labeling are not guided by such a theorem to consider a individually and locally enhanced model, and thus does not decouple an enhanced model from the predictive model to perform such a local enhancement.

Besides, different goals lead to different designs among our framework, self-training and auto-labeling. First, the goal of self-training is to annotate unlabeled data to further leverage them to train the classifier (Wei et al., 2021a). Hence, they focus on accuracy of the prediction made by the predictive model on the unlabeled data. However, the goal of our work is to build a framework to hand inaccurate supervision. Hence, we make more efforts on the reliability of the supervision information, and thus design global detection with the predictive model and local enhancement with another enhanced model, to further enhance the reliability of the supervision information. Second, the goal of auto-labeling is to obtain a labeled dataset (not a model) (Vishwakarma et al., 2023). Hence, they first train an annotation model and then select instances which the model could predict accurately. In contrast, our framework first finds the unreliable instances which the model could predict inaccurately, and then fine-tune an enhanced model to predict them accurately. This is because we aim to enhance the reliability of the supervision information and finally obtain a predictive model on unobserved instances, instead of only an annotation model on observed unlabeled instances.