



# Learning from Heterogeneous Label Quality: Strategies for Robust Model Training

Received: July 12, 2026

Revised: August 21, 2026

Accepted: September 14, 2026

Publish: September 28, 2026

Wu Shukun\*, Hafiz Muhammad Kurniawan, Adolf Asih Suprianto

## Abstract:

**Background:** Training data collected from crowdsourcing platforms, automated labeling systems, and multiple annotators often contain labels with unequal levels of reliability. Conventional machine-learning methods generally treat all labels as equally trustworthy, which can reduce model performance when label errors vary across samples and annotation sources.

**Aims:** This study proposes a quality-aware training method that explicitly models the heterogeneity of label reliability. It examines whether sample-level quality weighting can improve predictive performance and robustness under synthetic and real-world label-noise conditions.

**Methods:** The proposed method combines annotation or neighborhood agreement, loss-based confidence, and source-level reliability to estimate a continuous quality score for each training sample. The scores are incorporated into a weighted cross-entropy objective and periodically updated during training. Experiments were conducted on CIFAR-10, CIFAR-100, CIFAR-10N, and CIFAR-100N using ResNet-18. The method was compared with cross-entropy, label smoothing, generalized cross-entropy, Co-teaching, and DivideMix.

**Result:** Under high heterogeneous noise, the proposed method achieved accuracies of 83.21% on CIFAR-10 and 54.62% on CIFAR-100, compared with 73.48% and 45.92% for conventional cross-entropy. It also achieved the highest accuracy under the evaluated real-world noisy-label conditions. Ablation and sensitivity analyses showed that the combined quality indicators were complementary and that the method remained effective under moderate quality-estimation errors.

**Conclusion:** Explicitly modeling label reliability improves robustness when training data contain unequal annotation quality. The proposed method provides a practical and computationally efficient strategy for learning from crowdsourced, weakly supervised, and multi-source datasets.

**Keywords:** Heterogeneous label quality; Label noise; Machine learning; Robust model training; Quality-aware learning.

## 1. INTRODUCTION

Supervised machine learning relies on labeled data to learn the relationship between input features and target outcomes (Chase et al., 2022; Rashidi et al., 2021; Shetty et al., 2022). The reliability of this learning process, therefore, depends not only on the quantity and representativeness of the training data but also on the quality of their labels. In many practical quality of their

labels. In many practical applications, however, labels are collected from multiple sources, including domain experts, crowd workers, automated labeling systems, heuristic rules, historical databases, and weak classifiers. These sources may differ substantially in expertise, consistency, coverage, and susceptibility to error (Liu et al., 2023; Priestley et al., 2023; Ye et al., 2023a). Consequently, the resulting dataset may contain labels with varying levels of reliability rather than a simple division between entirely correct and entirely incorrect annotations. This condition is referred to in this study as heterogeneous label quality, where the probability that a label is correct varies across samples, annotators, data sources, or subsets of the training data (Gao et al., 2024; Klie et al., 2024; Q. Wang et al., 2021).

Heterogeneous label quality is increasingly important because large-scale data collection frequently prioritizes speed, coverage, and cost efficiency over exhaustive expert verification. Crowdsourcing and weak supervision can reduce annotation costs, but they also introduce disagreements, systematic biases, ambiguous

### Publisher Note:

CV Media Inti Teknologi stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



### Copyright

©2026 by the author(s).

Licensee CV Media Inti Teknologi, Indonesia. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution-ShareAlike (CCBY-SA) license (<https://creativecommons.org/licenses/by-sa/4.0/>).

annotations, and source-dependent errors. Weak-supervision systems such as Snorkel have demonstrated that multiple imperfect labeling sources can be combined by estimating their accuracies and dependencies, rather than treating all sources as equally reliable (Braun, 2024; Haliburton et al., 2025; Ratner et al., 2019; Yang et al., 2024). Similarly, learning from multiple external sources requires mechanisms that suppress corrupted or irrelevant information while retaining useful observations from partially reliable sources (Capstick et al., 2025; Kukreja et al., 2023; Liu & Letchmunan, 2024; Ye et al., 2023b). Recent research has also emphasized annotator agreement as an informative signal for identifying relatively reliable labels when multiple inaccurate annotations are available (Han et al., 2024; Lazaros et al., 2026; Pradhan et al., 2022). These developments indicate that label quality should be modeled as a graded, sample-dependent property rather than a uniform characteristic of an entire dataset.

Ignoring differences in label reliability can negatively affect model training. Highly flexible models may first learn regular patterns from correctly labeled samples but eventually memorize mislabeled observations, reducing their ability to generalize to unseen data (Bernhardt et al., 2022; de Vos et al., 2023; Higashimoto et al., 2024). Label errors may also distort model comparisons and produce misleading conclusions about the relative performance of learning algorithms, particularly when errors occur in benchmark evaluation data (Klie et al., 2023; Mohammed et al., 2025; Northcutt, Athalye, et al., 2021; Y. Wang et al., 2024). Furthermore, real-world label errors are generally more complex than uniformly generated synthetic noise. They may be associated with class ambiguity, feature similarity, annotation difficulty, or the characteristics of a particular labeling source. showed that controlled real-world label noise can differ meaningfully from conventional synthetic label corruption, supporting the need to evaluate robust learning methods under both simulated and naturally occurring noise conditions.

Several strategies have been developed to reduce the effects of unreliable labels. Noise-robust loss functions modify the learning objective so that incorrectly labeled samples have less influence on parameter estimation. For example, generalized cross-entropy loss balances the learning efficiency of categorical cross-entropy with the noise tolerance of mean absolute error (Chen et al., 2022; Gil-González et al., 2025; Xu et al., 2025; Z. Zhang & Sabuncu, 2018). Sample-selection methods, including Co-teaching, train multiple models that exchange samples considered likely to have correct labels (de Vos et al., 2023). Semi-supervised approaches such as DivideMix separate relatively clean and noisy samples and subsequently treat the latter as unlabeled data during training (T. Zhang et al., 2026). Other methods assign different weights to training examples. (Zhong et al., 2021), for instance, proposed learning example weights through meta-learning by using a small, clean validation set. Label-quality assessment techniques, such as confident learning, estimate the

relationship between observed and latent labels to identify probable annotation errors before or during model development (Northcutt et al., 2021b).

Although these methods have substantially advanced learning with noisy labels, several limitations remain when label quality is heterogeneous. First, many methods represent noise through a global noise rate, a class-level transition pattern, or a binary clean-versus-noisy partition. Such representations may not fully capture gradual differences in reliability among individual samples. Second, approaches that infer reliability primarily from training loss can mistakenly down-weight difficult but correctly labeled observations because high loss may indicate either label corruption or genuine classification difficulty. Third, meta-learning-based reweighting often relies on a trusted validation set, which may be unavailable or costly to construct. Fourth, methods designed for multiple annotators or weak-supervision sources often concentrate on producing an aggregated label. Although aggregation is useful, the estimated reliability associated with each resulting label is not always explicitly retained as a continuous training signal for the downstream predictive model. Finally, the effectiveness of quality-aware methods may depend on the accuracy of their label-quality estimates, yet sensitivity to imperfect confidence estimation is not consistently examined.

Therefore, an important research gap concerns how heterogeneous evidence about label reliability can be translated into a simple, general training mechanism that remains useful even when the estimated quality scores are not perfectly accurate. To address this gap, this study proposes a quality-aware training approach in which each training sample is assigned a confidence-based weight. Label confidence represents the estimated reliability of an observed label and is derived from available indicators of agreement and noise. Agreement indicators describe the consistency of a label with labels provided by other annotators or sources, whereas noise indicators capture evidence that an annotation may be uncertain, inconsistent, or corrupted. Instead of completely removing suspected samples or assigning equal importance to every observation, the proposed approach adjusts each sample's contribution to the learning objective based on its estimated reliability. This continuous weighting mechanism is intended to preserve potentially informative observations while limiting the influence of labels that are more likely to be incorrect.

The primary objective of this study is to determine whether explicitly incorporating heterogeneous label reliability into model training improves predictive performance and robustness compared with conventional training that assigns equal importance to all labeled samples. More specifically, the study investigates three research questions. First, to what extent does quality-aware sample weighting improve model performance under heterogeneous label quality? Second, how does the proposed approach behave across different noise levels and noise conditions, including

simulated and real label noise? Third, how sensitive is the approach to errors in the estimated label-confidence values? The investigation tests three corresponding hypotheses: H1, quality-aware training yields better predictive performance than conventional unweighted training; H2, quality-aware training maintains greater robustness as label noise and quality heterogeneity increase; and H3, the performance advantage remains observable even with moderate errors in label-quality estimation.

A comparative experimental design is employed using benchmark classification datasets containing simulated and real label noise. The principal independent variables are the training strategy, label-noise condition, level of label-quality heterogeneity, and degree of error introduced into the estimated confidence scores. The dependent variables are predictive accuracy and robustness under changing label-quality conditions. Conventional model training is used as the principal baseline, while the proposed method incorporates sample-specific confidence weights into the training objective. Sensitivity experiments systematically perturb the estimated quality scores to determine whether the method remains reliable when label confidence cannot be measured precisely.

The novelty of this study lies in its integration of heterogeneous label-quality estimation, continuous sample weighting, and sensitivity analysis within a unified training framework. Unlike methods that assume uniform noise, discard uncertain observations, or require perfectly estimated label reliability, the proposed strategy treats label quality as a continuum and directly incorporates it into model optimization. The study contributes an adaptable mechanism for training models with imperfect labels, an empirical comparison between simulated and real-world noise settings, and evidence on the consequences of label-quality misestimation. These contributions are particularly relevant to crowdsourced, weakly supervised, and multi-source datasets, where

label reliability is naturally uneven and obtaining a completely verified training set is often impractical.

## 2. MATERIAL AND METHOD

### 2.1 Research Design

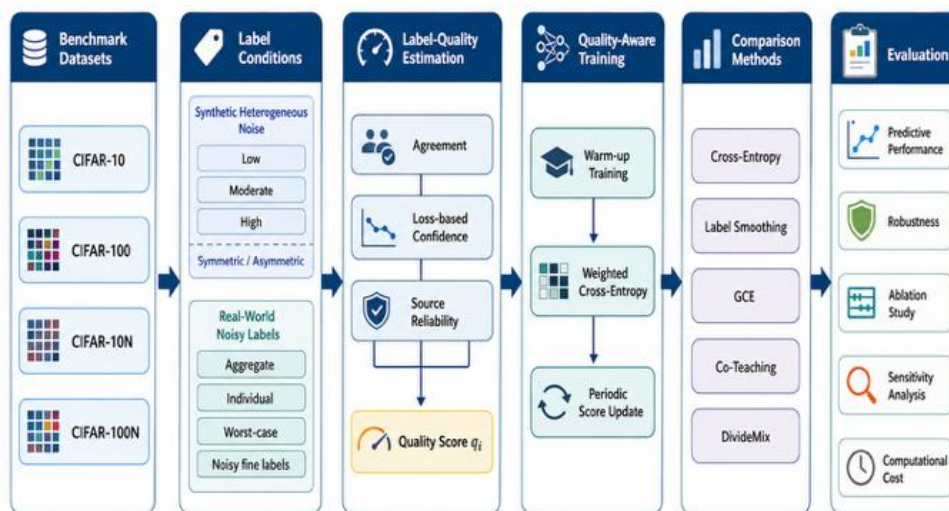
This study employed a controlled comparative experimental design to evaluate the effectiveness of quality-aware training under heterogeneous label-quality conditions (Özdemir et al., 2026). Heterogeneous label quality refers to a condition in which the probability of label correctness differs across samples or annotation sources (Herde et al., 2021). The proposed method estimated a continuous reliability score for each training label and used the score to control the contribution of the corresponding sample during model optimization.

The experiment consisted of six stages:

- preparation of benchmark datasets;
- construction of synthetic heterogeneous label-quality conditions;
- preparation of real-world noisy-label conditions;
- estimation of sample-level label reliability;
- quality-aware model training and baseline comparison; and
- robustness, ablation, and sensitivity evaluation.

The proposed method was evaluated under both controlled synthetic noise and naturally occurring human annotation errors. Synthetic experiments revealed the true noise rate and label correctness, whereas real-world noisy-label experiments better reflected more realistic annotation behaviour.

The overall experimental procedure is summarized in Figure 1. The workflow integrates dataset preparation, heterogeneous label-noise construction, label-quality estimation, quality-aware optimization, baseline comparison, and multidimensional performance evaluation.



**Figure 1.** Experimental workflow of the proposed quality-aware training method

As illustrated in Figure 1, the framework begins with benchmark datasets containing either synthetically generated or human-produced noisy labels. Agreement, loss-based confidence, and source reliability are then combined to estimate a continuous quality score for each training sample. These scores determine the contribution of individual samples to the weighted training objective. The trained model is subsequently compared with conventional and noise-robust baselines using predictive, robustness, sensitivity, ablation, and computational measures.

## 2.2 Datasets

The experiments used CIFAR-10 and CIFAR-100 as the primary image-classification benchmarks (Brigato et al., 2022). CIFAR-10 consists of 60,000 colour images distributed across 10 classes, whereas CIFAR-100 contains the same number of images distributed across 100 fine-grained classes. Each dataset contains 50,000 training images and 10,000 test images with a resolution of  $32 \times 32$  pixels (Hafiz et al., 2022; Zhao et al., 2022).

The original CIFAR training labels were treated as clean reference labels when generating synthetic noisy-label conditions. The original test sets were unmodified and used only for final model evaluation.

Real-world noisy-label experiments used CIFAR-10N and CIFAR-100N. These datasets contain human-generated annotations for the original CIFAR images. CIFAR-10N provides multiple human annotation variants, including individual, aggregated, and worst-case labels. CIFAR-100N provides human-generated noisy fine-grained labels for CIFAR-100. These datasets were selected because their annotation errors are more realistic than uniformly generated synthetic noise (González-Santoyo et al., 2025; Song et al., 2025).

The datasets used in this study are summarized in Table 1. The selection includes standard benchmark datasets with verified labels and their corresponding human-annotated noisy-label versions, thereby supporting evaluation under both controlled and realistic label-noise conditions

**Table 1.** Datasets Used in the Experimental Evaluation

| Dataset    | Number of Classes | Training Samples | Test Samples | Label Condition              | Experimental Role                         |
|------------|-------------------|------------------|--------------|------------------------------|-------------------------------------------|
| CIFAR-10   | 10                | 50,000           | 10,000       | Clean reference labels       | Synthetic heterogeneous-noise experiments |
| CIFAR-100  | 100               | 50,000           | 10,000       | Clean reference labels       | Synthetic heterogeneous-noise experiments |
| CIFAR-10N  | 10                | 50,000           | 10,000       | Human-generated noisy labels | Real-world label-noise experiments        |
| CIFAR-100N | 100               | 50,000           | 10,000       | Human-generated noisy labels | Real-world label-noise experiments        |

As shown in Table 1, CIFAR-10 and CIFAR-100 were used to construct controlled synthetic heterogeneous-noise conditions, while CIFAR-10N and CIFAR-100N were used to evaluate performance under human-generated annotation errors. The datasets differ mainly in classification complexity and label condition. CIFAR-10 contains 10 classes, whereas CIFAR-100 contains 100 fine-grained classes, making the latter a more challenging benchmark. Using both clean-reference and human-annotated datasets enabled the proposed method to be assessed under known corruption patterns and more realistic instance-dependent label errors.

For each dataset, the original training set was divided using stratified sampling into 45,000 training samples and 5,000 validation samples. The official test set remained unchanged. The same partitions were used for all compared methods.

In the real-world noisy-label experiments, the original verified CIFAR labels were hidden from all training and label-quality estimation procedures. They were accessed only after training to evaluate whether the estimated quality scores correctly identified unreliable labels.

## 2.3 Data Preprocessing

All images were converted into floating-point tensors and normalized using the channel-wise mean and standard deviation calculated from the corresponding training set.

During training, the following augmentation procedures were applied:

- random horizontal flipping with a probability of 0.5;
- four-pixel reflection padding; and
- random cropping to  $32 \times 32$  pixels.

No augmentation was applied to the validation and test sets. All compared methods used exactly the same preprocessing and augmentation procedures.

The training-set statistics were used for normalization to prevent the validation or test data from influencing the model development process.

## 2.4 Construction of Heterogeneous Label-Quality Conditions

### 2.4.1 Source Assignment

To simulate data collected from sources with different annotation reliability, the CIFAR-10 and CIFAR-100

training samples were divided into four equally sized virtual annotation sources:

$$\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2 \cup \mathcal{D}_3 \cup \mathcal{D}_4. \quad (1)$$

Source assignment was stratified by class to ensure that each source contained approximately the same class distribution. Each sample belonged to only one source.

The four sources represented highly reliable, moderately reliable, weakly reliable, and highly unreliable annotation processes.

#### 2.4.2 Heterogeneous Noise Profiles

Table 2 presents the source-specific corruption rates used to simulate heterogeneous label quality. Three profiles were defined to represent progressively greater differences in annotation reliability among the four virtual data sources.

**Table 2.** Source-Specific Label-Noise Profiles

| Quality Profile        | Source 1 | Source 2 | Source 3 | Source 4 | Average Noise Rate |
|------------------------|----------|----------|----------|----------|--------------------|
| Low heterogeneity      | 5%       | 10%      | 15%      | 20%      | 12.50%             |
| Moderate heterogeneity | 5%       | 15%      | 30%      | 45%      | 23.75%             |
| High heterogeneity     | 10%      | 30%      | 50%      | 70%      | 40.00%             |

As shown in Table 2, the low-heterogeneity profile assigned corruption rates ranging from 5% to 20%, producing an average noise rate of 12.50%. Under moderate heterogeneity, the rates increased from 5% for the most reliable source to 45% for the least reliable source, resulting in an average of 23.75%. The high-heterogeneity profile was the most challenging condition, with source-specific rates ranging from 10% to 70% and an average corruption rate of 40.00%. These profiles allowed the experiment to examine whether model performance was affected not only by the overall amount of label noise but also by unequal reliability across annotation sources.

For the sample  $i$  belonging to the source  $s_i$ , the observed label was corrupted according to the source-specific probability  $\rho_{s_i}$ :

$$P(\tilde{y}_i \neq y_i | s_i) = \rho_{s_i}, \quad (2)$$

where  $y_i$  denotes the clean reference label and  $\tilde{y}_i$  denotes the observed training label.

Uniform noise control conditions were also generated with overall corruption rates of 12.50%, 23.75%, and 40.00%. These rates matched the average corruption rates of the heterogeneous conditions. The comparison made it possible to determine whether performance was influenced only by the total amount of label noise or also by unequal label quality across sources.

#### 2.4.3 Symmetric and Asymmetric Noise

Two label-corruption mechanisms were considered.

In the symmetric-noise condition, a corrupted label was replaced uniformly with one of the remaining class labels:

$$P(\tilde{y}_i = j | y_i = k) = \frac{\rho_{s_i}}{C-1}, j \neq k, \quad (3)$$

where  $C$  represents the number of classes.

In the asymmetric-noise condition, labels were changed to semantically related classes. For CIFAR-10, the following mappings were used:

- truck  $\rightarrow$  automobile;
- bird  $\rightarrow$  airplane;

- deer  $\rightarrow$  horse;
- cat  $\rightarrow$  dog; and
- dog  $\rightarrow$  cat.

For CIFAR-100, label changes were restricted to classes belonging to the same superclass. A corrupted label was replaced by the next fine-grained class within its superclass using a fixed cyclic mapping. The mapping was generated once and retained across all experimental repetitions.

Each synthetic condition was independently generated for every random seed.

#### 2.5 Real-World Noisy-Label Conditions

The CIFAR-10N experiments used three annotation conditions:

- aggregate human labels;
- randomly selected individual human labels; and
- worst-case human labels.

The aggregate condition represented labels selected from the majority of human annotations. The individual condition represented labels obtained from a single annotator. The worst-case condition contained the most challenging human annotations available in CIFAR-10N.

For CIFAR-100N, the provided human-generated noisy fine labels were used as the observed training labels.

When multiple human annotations were available for a sample, annotator agreement was calculated directly. When only one noisy annotation was available, agreement was estimated from neighbouring samples in the learned feature space, as described in Section 2.6.

No synthetic corruption was added to CIFAR-10N or CIFAR-100N.

#### 2.6 Estimation of Label Quality

The proposed approach estimated a quality score  $q_i \in [0,1]$  for every training sample. A high value indicated that the observed label was likely to be reliable, whereas a low value indicated that it was uncertain or potentially incorrect.

The quality score combined three signals:

- annotation or neighbourhood agreement;
- normalized sample-loss confidence; and
- source-level reliability.

### 2.6.1 Warm-Up Training

Before estimating label quality, the model was trained for 10 warm-up epochs using conventional cross-entropy. All samples received equal weights during this period.

The warm-up stage allowed the model to learn basic class structures before sample reliability was estimated. The loss of every sample was recorded at each warm-up epoch.

The exponential moving average of the sample loss was calculated as

$$\bar{\ell}_i^{(t)} = \lambda \bar{\ell}_i^{(t-1)} + (1 - \lambda) \ell_i^{(t)}, \quad (4)$$

where  $\ell_i^{(t)}$  is the cross-entropy loss of the sample  $i$  at epoch  $t$ , and  $\lambda = 0.8$ .

### 2.6.2 Annotation and Neighbourhood Agreement

When multiple annotations were available, direct annotation agreement was calculated as

$$A_i = \frac{1}{M_i} \sum_{m=1}^{M_i} \mathbb{I}(y_{im} = \tilde{y}_i), \quad (5)$$

where  $M_i$  is the number of annotations for the sample  $i$ ,  $y_{im}$  is the label provided by the annotator  $m$ , and  $\mathbb{I}(\cdot)$  is the indicator function.

When only one annotation was available, neighbourhood agreement was estimated from the model's penultimate-layer representation. The 20 nearest training samples were identified using cosine distance. The agreement score was then calculated as

$$A_i = \frac{1}{K} \sum_{j \in \mathcal{N}_K(i)} \mathbb{I}(\tilde{y}_j = \tilde{y}_i), \quad (6)$$

where  $K = 20$  and  $\mathcal{N}_K(i)$  denotes the nearest-neighbour set of samples  $i$ .

The neighbour search was conducted only within the training partition.

### 2.6.3 Loss-Based Confidence

Samples with persistently high losses relative to other samples in the same observed class were considered more likely to have unreliable labels.

The loss-based confidence score was calculated from the within-class percentile rank:

$$C_i = 1 - \text{PercentileRank}(\bar{\ell}_i | \tilde{y}_i). \quad (7)$$

A value close to 1 indicated a relatively low loss, whereas a value close to 0 indicated a persistently high loss.

Within-class ranking was used instead of global loss normalization because some classes were naturally more difficult to classify than others.

### 2.6.4 Source Reliability

For datasets containing source information, the reliability of the source  $s$  was estimated as

$$R_s = \frac{1}{|\mathcal{D}_s|} \sum_{i \in \mathcal{D}_s} A_i C_i. \quad (8)$$

This score represented the average agreement and loss-based confidence of the samples obtained from a particular source.

For datasets without source metadata, the source component was omitted, and the remaining component weights were renormalized.

### 2.6.5 Final Quality Score

When source metadata were available, the sample-level quality score was calculated as

$$\hat{q}_i = 0.40A_i + 0.40C_i + 0.20R_{s_i}. \quad (9)$$

When source metadata were unavailable, the score was calculated as

$$\hat{q}_i = 0.50A_i + 0.50C_i. \quad (10)$$

The score was clipped to the interval  $[0.05, 1.00]$ :

$$\hat{q}_i = \min(1, \max(0.05, \hat{q}_i)). \quad (11)$$

A minimum value of 0.05 was used so that potentially mislabeled samples were not completely discarded.

After the warm-up stage, quality scores were recalculated every five epochs. To reduce instability, each update used exponential smoothing:

$$q_i^{(t)} = 0.8q_i^{(t-1)} + 0.2\hat{q}_i^{(t)}. \quad (12)$$

### 2.7 Quality-Aware Model Training

The estimated quality score was converted into a training weight:

$$w_i = q_i^\tau, \quad (13)$$

where the sharpening parameter was set to  $\tau = 2$ . This transformation increased the difference between highly reliable and uncertain labels while preserving continuous weights.

The proposed quality-aware loss for the mini-batch  $\mathcal{B}$  was defined as

$$\mathcal{L}_{QA} = \frac{\sum_{i \in \mathcal{B}} w_i [-\log p_\theta(\tilde{y}_i | x_i)]}{\sum_{i \in \mathcal{B}} w_i}, \quad (14)$$

where  $p_\theta(\tilde{y}_i | x_i)$  represents the probability assigned by the model  $\theta$  to the observed label.

The weights in every mini-batch were normalized to have a mean of 1:

$$w_i^* = \frac{w_i}{\frac{1}{|\mathcal{B}|} \sum_{j \in \mathcal{B}} w_j}. \quad (15)$$

This normalization prevented the average gradient magnitude from being incorrectly interpreted as an improvement due to quality-aware weighting.

The complete training procedure is summarized in Algorithm 1.

#### Algorithm 1. Quality-Aware Training

Input: Training data  $\{x_i, \tilde{y}_i, s_i\}_{i=1}^N$ , number of epochs  $T$ , warm-up epochs  $T_w = 10$

Output: Trained model parameters  $\theta$

1. Initialize model parameters  $\theta$ .

2. Train the model using ordinary cross-entropy for  $T_w$  epochs.
3. Record the exponential moving average of each sample's loss.
4. Extract the penultimate-layer feature representation.
5. Calculate annotation or neighbourhood agreement  $A_i$ .
6. Calculate loss-based confidence  $C_i$ .
7. Calculate source reliability  $R_s$ , when available.
8. Calculate and clip the quality score  $q_i$ .
9. For epochs  $T_w + 1$  to  $T$ :
  - a. construct mini-batches using random sampling;
  - b. calculate normalized weights  $w_i^*$ ;
  - c. optimize the quality-aware loss;
  - d. update the quality scores every five epochs.
10. Return the final model parameters.

Algorithm 1 describes the complete quality-aware training process. The model is first trained for 10 warm-up epochs using standard cross-entropy, with each sample's loss recorded. Afterward, feature representations, annotation or neighbourhood agreement, loss-based confidence, and source reliability are used to estimate the quality score  $q_i$  for each sample.

During the main training stage, these quality scores are converted into normalized weights  $w_i^*$ . Samples with more reliable labels receive greater influence, while uncertain samples receive smaller but nonzero weights. The model is then optimized using the quality-aware loss, and the quality scores are updated every five epochs. The final trained model parameters are returned after all  $T$  epochs are completed.

### 2.8 Model Architecture and Training Configuration

A ResNet-18 model was used as the common classifier for all experiments. The architecture was adapted for  $32 \times 32$  pixel images by replacing the original  $7 \times 7$  input convolution with a  $3 \times 3$  convolution using a stride of 1. The initial max-pooling layer was removed.

The final classification layer contained 10 output units for CIFAR-10 and CIFAR-10N and 100 output units for CIFAR-100 and CIFAR-100N.

The model architecture and optimization parameters used consistently across the experiments are reported in Table 3. A common configuration was maintained wherever method-specific requirements permitted to support a fair and reproducible comparison.

**Table 3.** Model-Training Configuration

| Parameter                     | Value                       |
|-------------------------------|-----------------------------|
| Model architecture            | ResNet-18                   |
| Total training epochs         | 200                         |
| Warm-up epochs                | 10                          |
| Batch size                    | 128                         |
| Optimizer                     | Stochastic gradient descent |
| Initial learning rate         | 0.1                         |
| Momentum                      | 0.9                         |
| Weight decay                  | $5 \times 10^{-4}$          |
| Learning-rate schedule        | Cosine annealing            |
| Minimum learning rate         | $1 \times 10^{-5}$          |
| Quality-score update interval | Every 5 epochs              |
| Number of nearest neighbours  | 20                          |
| Minimum sample weight         | 0.05                        |
| Weight-sharpening parameter   | 2                           |
| Independent repetitions       | 5                           |

As summarized in Table 3, all experiments used an adapted ResNet-18 architecture trained for 200 epochs with stochastic gradient descent. The model was trained using a batch size of 128, an initial learning rate of 0.1, momentum of 0.9, and weight decay of  $5 \times 10^{-4}$ . A cosine-annealing schedule reduced the learning rate to  $1 \times 10^{-5}$ . The first 10 epochs were used for warm-up training, after which quality scores were updated every five epochs. Neighbourhood agreement was calculated using 20 nearest neighbours, while the minimum sample weight and weight-sharpening parameter were set to 0.05 and 2, respectively. Each experimental condition was repeated 5 times to account for variation due to random initialization and noise generation.

The learning rate at epoch  $t$  was determined using cosine annealing:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min}) \left[ 1 + \cos \left( \frac{\pi t}{T} \right) \right], \quad (16)$$

where  $\eta_{\max} = 0.1$ ,  $\eta_{\min} = 10^{-5}$ , and  $T = 200$ .

The model obtained at the final training epoch was used for test-set evaluation. Test-set results were not used for checkpoint selection or hyperparameter tuning.

### 2.9 Comparison Methods

The proposed quality-aware approach was compared with the following methods.

#### 2.9.1 Conventional Cross-Entropy

All training samples were assigned equal importance. This method represented the standard supervised-learning baseline:

$$\mathcal{L}_{CE} = -\frac{1}{|\mathcal{B}|} \sum_{i \in \mathcal{B}} \log p_{\theta}(\tilde{y}_i | x_i). \quad (17)$$

### 2.9.2 Label Smoothing

Label smoothing was applied using a smoothing coefficient of 0.1. This baseline examined whether reducing confidence in the observed labels was sufficient to improve robustness.

### 2.9.3 Generalized Cross-Entropy

Generalized cross-entropy was included as a noise-robust loss-function baseline. Its robustness parameter was set to  $q = 0.7$ , following the commonly used configuration proposed by Zhang & Sabuncu (2018).

### 2.9.4 Co-Teaching

Co-teaching trained two independently initialized ResNet-18 models. During each mini-batch, each model selected its low-loss samples and passed them to the other model for parameter updates. The sample retention rate was gradually reduced over the first 30 epochs based on the estimated overall noise rate.

### 2.9.5 DivideMix

DivideMix was included as a representative semi-supervised noisy-label method. It fitted a two-component Gaussian mixture model to the sample-loss distribution and divided the training data into probable clean and noisy groups. The probable noisy samples were subsequently treated as unlabeled data.

For fairness, all methods used the same dataset partitions, image preprocessing, augmentation, ResNet-18 backbone, training epochs, optimizer family, and random seeds wherever compatible with the original algorithm.

## 2.10 Evaluation Metrics

### 2.10.1 Predictive Performance

Model performance was evaluated using:

- classification accuracy;
- macro-averaged precision;
- macro-averaged recall;
- macro-averaged F1-score; and
- balanced accuracy.

Macro-averaged measures were calculated independently for each class and then averaged:

$$F1_{\text{macro}} = \frac{1}{C} \sum_{c=1}^C F1_c. \quad (18)$$

These measures were included because overall accuracy may conceal poor performance in difficult classes, particularly in CIFAR-100.

### 2.10.2 Robustness to Label Noise

Relative performance degradation was used to quantify robustness:

$$RD = \frac{M_{\text{clean}} - M_{\text{noisy}}}{M_{\text{clean}}} \times 100\%, \quad (19)$$

where  $M_{\text{clean}}$  is the metric obtained under clean-label training, and  $M_{\text{noisy}}$  is the corresponding metric under a noisy-label condition.

A smaller value of  $RD$  indicated greater robustness.

The improvement of the proposed method over conventional cross-entropy was calculated as

$$\Delta M = M_{\text{QA}} - M_{\text{CE}}, \quad (20)$$

where  $M_{\text{QA}}$  and  $M_{\text{CE}}$  denote the metric values of quality-aware and conventional training, respectively.

### 2.10.3 Label-Quality Estimation

The ability of the proposed quality score to distinguish correct from incorrect labels was evaluated using:

area under the receiver operating characteristic curve;

area under the precision-recall curve;

Spearman rank correlation; and

mean quality-score separation.

Quality-score separation was calculated as

$$QS = \bar{q}_{\text{correct}} - \bar{q}_{\text{incorrect}}. \quad (21)$$

A larger positive value indicated that the estimator placed greater confidence in correct labels than in corrupted ones.

The verified training labels were used only for this post hoc evaluation and were never provided to the training algorithm.

### 2.10.4 Computational Efficiency

Computational requirements were measured using:

average training time per epoch;

total training time;

peak GPU memory consumption; and

computational overhead relative to conventional cross-entropy.

The percentage overhead was calculated as

$$CO = \frac{T_{\text{method}} - T_{\text{CE}}}{T_{\text{CE}}} \times 100\%. \quad (22)$$

## 2.11 Ablation Study

An ablation study was conducted to determine the contribution of each component of the quality-estimation mechanism.

The complete method was compared with the following variants:

- agreement only, using  $q_i = A_i$ ;
- loss of confidence only, using  $q_i = C_i$ ;
- source reliability only, using  $q_i = R_{S_i}$ ;
- agreement and loss of confidence, without source reliability;
- static quality scores, calculated once after the warm-up period; and
- complete quality-aware method, using all available components and periodic updates.

The same model architecture, optimizer, data partitions, and random seeds were used across all ablation conditions.

The ablation experiment was conducted under the moderate and high heterogeneous-noise profiles because these settings provided sufficient variation in label quality to reveal differences among the components.

### 2.12 Sensitivity to Quality-Estimation Errors

The proposed method may be applied in situations where label reliability cannot be estimated perfectly. Therefore, two sensitivity experiments were conducted.

#### 2.12.1 Additive Perturbation

Zero-mean Gaussian perturbations were added to the estimated quality scores:

$$q'_i = \text{clip}(q_i + \epsilon_i, 0.05, 1), \epsilon_i \sim \mathcal{N}(0, \sigma^2). \quad (23)$$

The evaluated standard deviations were:

$$\sigma \in \{0.05, 0.10, 0.20, 0.30\}.$$

This experiment simulated increasingly inaccurate confidence estimates.

#### 2.12.2 Quality-Ranking Perturbation

Quality scores were randomly permuted among selected proportions of the training samples. The evaluated proportions were 10%, 20%, 30%, and 40%.

This procedure simulated situations in which the method incorrectly ranked unreliable labels as reliable and reliable labels as unreliable.

Sensitivity was measured as the difference between performance with perturbed and unperturbed quality scores:

$$SD = M_{\text{perturbed}} - M_{\text{original}}. \quad (24)$$

A value close to zero indicated resilience to label-quality estimation errors.

### 2.13 Statistical Analysis

Each experimental configuration was repeated five times using the random seeds 11, 23, 37, 51, and 71. Results were reported as mean values and standard deviations.

The Friedman test was used to examine whether statistically significant differences existed among the compared methods across matched datasets and noise conditions. When the Friedman test was significant, the proposed method was compared separately with each baseline using two-sided Wilcoxon signed-rank tests.

Holm's correction was applied to control the family-wise error rate caused by multiple pairwise comparisons. Statistical significance was determined at

$$p < 0.05.$$

Rank-biserial correlation was reported as the effect-size measure for pairwise comparisons. Effect sizes were interpreted as negligible, small, moderate, or large based on their absolute values.

In addition, 95% confidence intervals for mean performance differences were estimated using 10,000 bootstrap resamples.

### 2.14 Implementation and Reproducibility

The experiment was implemented using Python 3.10 and PyTorch 2.2.1. Image processing and model architectures were implemented using Torchvision 0.17.1. Numerical analysis was performed using NumPy 1.26, while statistical analyses were performed using SciPy 1.12 and Statsmodels 0.14.

Training was conducted on a workstation equipped with:

- an NVIDIA RTX 3090 GPU with 24 GB memory;
- an Intel Core i9 processor;
- 64 GB system memory; and
- Ubuntu 22.04.

To improve reproducibility, random seeds were assigned to Python, NumPy, and PyTorch. Deterministic CUDA operations were enabled whenever supported. The training-data partitions, virtual-source assignments, synthetic noise masks, and asymmetric class mappings were saved before model training and reused by all compared methods.

The same configuration files were used across all independent repetitions. The implementation recorded the following information for every run:

- random seed;
- dataset partition;
- label-noise profile;
- source assignment;
- training and validation loss;
- estimated sample-quality scores;
- final test metrics;
- training time; and
- peak memory consumption.

All hyperparameters were fixed before the final evaluation. No hyperparameter was selected using the official test set.

### 2.15 Methodological Validity and Limitations

Several procedures were used to support internal validity. All compared methods used identical training-test partitions, preprocessing operations, model backbones, and random seeds. The official test sets remained isolated from training and quality estimation. Clean training labels were used only to create controlled synthetic conditions and to evaluate label-quality estimation after training.

The comparison between heterogeneous and uniform-noise conditions, with the average noise rate controlled. Therefore, performance differences between these conditions could be attributed more directly to unequal label reliability rather than merely to a larger number of incorrect labels.

The methodology nevertheless has several limitations. The experiments were restricted to closed-set, single-label image classification. They did not evaluate multilabel classification, regression, open-set noise, or simultaneous corruption of input features and labels. The virtual annotation sources used in the synthetic experiments represented controlled approximations of real data providers. In practical applications, source

reliability may change over time or differ across classes. Furthermore, the neighbourhood-agreement estimator may be less reliable in the early stages of training, when the learned feature representation is incomplete. These limitations were considered when interpreting the generalizability of the results.

### 3. RESULT AND DISCUSSION

#### 3.1 Results

The experimental results were analyzed from five perspectives: predictive performance under synthetic heterogeneous noise, performance under real-world

noisy labels, label-quality estimation accuracy, component ablation, and sensitivity to quality-estimation errors. All values are reported as the mean and standard deviation of five independent runs.

#### Performance Under Synthetic Heterogeneous Label Noise

Table 4 compares the classification accuracy and macro-F1 score of the six training methods under clean, low-, moderate-, and high-heterogeneity conditions. The results are reported as the mean and standard deviation of five independent runs, followed by the corresponding macro-F1 score.

**Table 4.** Predictive Performance Under Synthetic Heterogeneous Label Noise

| Dataset and Condition             | Cross-Entropy            | Label Smoothing          | GCE                      | Co-Teaching              | DivideMix                | Proposed Method          |
|-----------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| CIFAR-10, clean                   | 93.18 $\pm$ 0.18 / 93.14 | 93.32 $\pm$ 0.20 / 93.28 | 92.84 $\pm$ 0.25 / 92.80 | 92.76 $\pm$ 0.28 / 92.72 | 93.41 $\pm$ 0.17 / 93.36 | 93.27 $\pm$ 0.19 / 93.23 |
| CIFAR-10, low heterogeneity       | 89.62 $\pm$ 0.35 / 89.55 | 90.14 $\pm$ 0.31 / 90.07 | 90.86 $\pm$ 0.29 / 90.78 | 91.02 $\pm$ 0.33 / 90.94 | 91.58 $\pm$ 0.27 / 91.51 | 92.03 $\pm$ 0.24 / 91.96 |
| CIFAR-10, moderate heterogeneity  | 84.06 $\pm$ 0.48 / 83.89 | 85.11 $\pm$ 0.44 / 84.94 | 86.72 $\pm$ 0.39 / 86.55 | 87.39 $\pm$ 0.42 / 87.20 | 88.63 $\pm$ 0.34 / 88.46 | 89.52 $\pm$ 0.31 / 89.35 |
| CIFAR-10, high heterogeneity      | 73.48 $\pm$ 0.72 / 72.96 | 75.21 $\pm$ 0.65 / 74.72 | 78.64 $\pm$ 0.58 / 78.19 | 79.85 $\pm$ 0.61 / 79.42 | 81.93 $\pm$ 0.49 / 81.51 | 83.21 $\pm$ 0.46 / 82.84 |
| CIFAR-100, clean                  | 74.62 $\pm$ 0.29 / 74.10 | 75.03 $\pm$ 0.27 / 74.50 | 73.81 $\pm$ 0.34 / 73.24 | 73.46 $\pm$ 0.38 / 72.92 | 75.21 $\pm$ 0.25 / 74.72 | 74.96 $\pm$ 0.27 / 74.45 |
| CIFAR-100, low heterogeneity      | 68.93 $\pm$ 0.44 / 68.15 | 69.74 $\pm$ 0.41 / 68.96 | 70.66 $\pm$ 0.39 / 69.91 | 70.98 $\pm$ 0.43 / 70.21 | 71.83 $\pm$ 0.36 / 71.08 | 72.41 $\pm$ 0.33 / 71.69 |
| CIFAR-100, moderate heterogeneity | 60.84 $\pm$ 0.61 / 59.77 | 62.01 $\pm$ 0.58 / 60.94 | 63.42 $\pm$ 0.52 / 62.39 | 64.11 $\pm$ 0.55 / 63.06 | 65.17 $\pm$ 0.47 / 64.13 | 66.03 $\pm$ 0.45 / 65.02 |
| CIFAR-100, high heterogeneity     | 45.92 $\pm$ 0.86 / 44.38 | 47.71 $\pm$ 0.79 / 46.22 | 50.21 $\pm$ 0.72 / 48.74 | 51.84 $\pm$ 0.75 / 50.39 | 53.38 $\pm$ 0.65 / 51.96 | 54.62 $\pm$ 0.61 / 53.24 |

Table 4 shows that the proposed method achieved performance comparable to the strongest baselines under clean-label conditions and produced the best results across all heterogeneous-noise conditions. On CIFAR-10, accuracy decreased from 93.27% with clean labels to 83.21% with high heterogeneity. In comparison, conventional cross-entropy decreased from 93.18% to 73.48%. Thus, under the most challenging conditions, the proposed method achieved a 9.73 percentage-point improvement over conventional training.

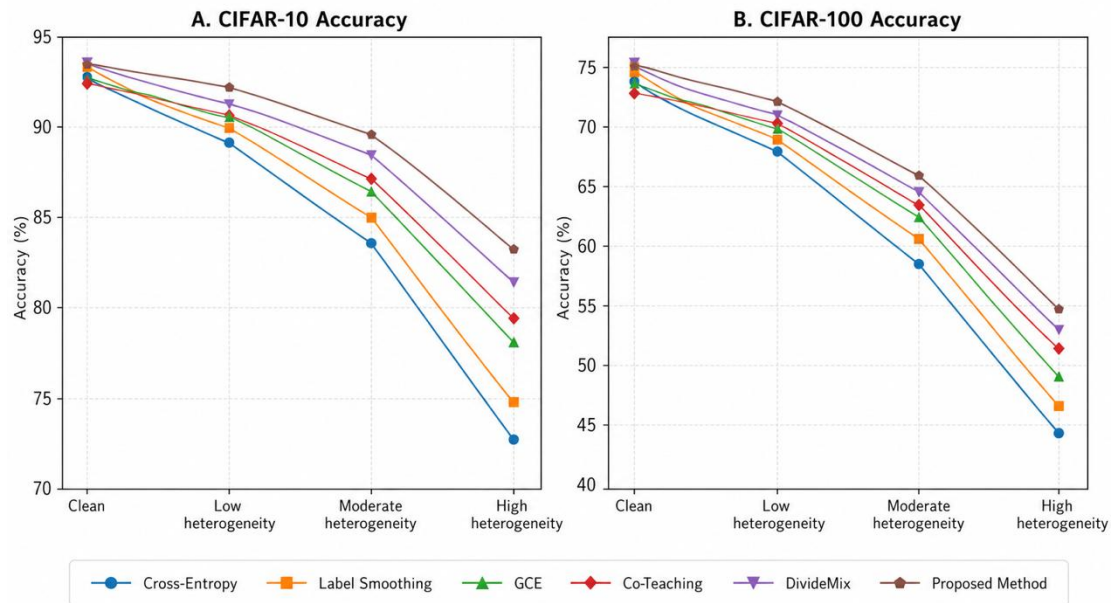
A similar pattern was observed on CIFAR-100. Under high heterogeneity, the proposed method obtained an accuracy of 54.62%, compared with 45.92% for cross-entropy and 53.38% for DivideMix. The macro-F1 results followed the same trend, with the proposed method achieving 82.84% on CIFAR-10 and 53.24% on CIFAR-100 under high heterogeneity. These findings demonstrate that quality-aware weighting reduced performance degradation across both overall accuracy and class-balanced performance.

The relative performance degradation of conventional cross-entropy from clean to high heterogeneous noise

was 21.14% on CIFAR-10. In comparison, the proposed method experienced a 10.79% degradation. On CIFAR-100, the corresponding degradation values were 38.46% and 27.13%. Therefore, the proposed method retained a larger proportion of its clean-label performance as label quality deteriorated.

The experiment also compared heterogeneous and uniform noise at matched average corruption rates. At an average noise rate of 23.75%, conventional cross-entropy achieved 86.31% accuracy under uniform CIFAR-10 noise but only 84.06% under heterogeneous noise. The proposed approach obtained 89.84% and 89.52%, respectively. This smaller difference suggests that quality-aware training was less sensitive to unequal noise distributions across annotation sources.

Although Table 4 provides detailed numerical results, the effect of increasing label-quality heterogeneity is more clearly observed in the performance trends shown in Figure 2. The figure compares the accuracy of the six training strategies across clean, low-, moderate-, and high-heterogeneity conditions.



**Figure 2.** Accuracy trends under increasing heterogeneous label noise

Figure 2 shows that all methods experienced a decline in accuracy as label-quality heterogeneity increased. However, the proposed method exhibited the smallest overall decline. Under high heterogeneity, it retained accuracies of 83.21% on CIFAR-10 and 54.62% on CIFAR-100, outperforming conventional cross-entropy by 9.73 and 8.70 percentage points, respectively. The widening performance gap indicates that the advantage of quality-aware weighting becomes more pronounced as label reliability becomes increasingly uneven.

#### Performance Under Real-World Noisy Labels

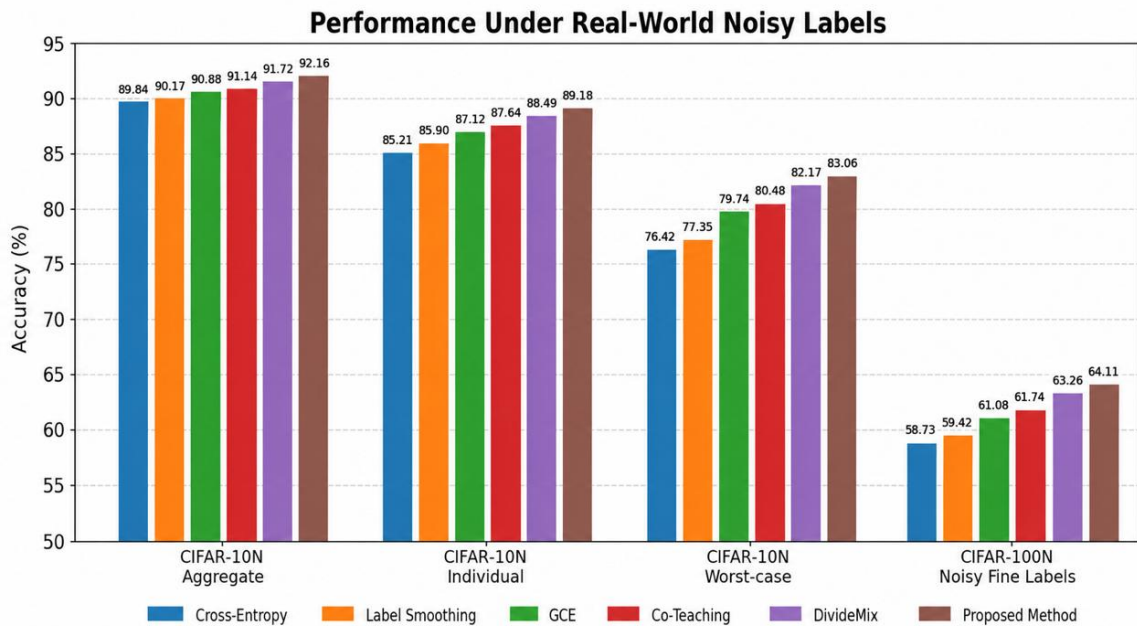
Table 5 reports the results obtained using human-generated annotations from CIFAR-10N and CIFAR-100N. The proposed method achieved the highest mean accuracy in all four real-world conditions.

**Table 5.** Performance Under Real-World Noisy Labels

| Dataset and Label Condition   | Cross-Entropy | Label Smoothing | GCE          | Co-Teaching  | DivideMix    | Proposed Method |
|-------------------------------|---------------|-----------------|--------------|--------------|--------------|-----------------|
| CIFAR-10N, aggregate          | 89.84 ± 0.33  | 90.17 ± 0.31    | 90.88 ± 0.29 | 91.14 ± 0.32 | 91.72 ± 0.27 | 92.16 ± 0.25    |
| CIFAR-10N, individual         | 85.21 ± 0.46  | 85.90 ± 0.43    | 87.12 ± 0.39 | 87.64 ± 0.42 | 88.49 ± 0.36 | 89.18 ± 0.34    |
| CIFAR-10N, worst-case         | 76.42 ± 0.68  | 77.35 ± 0.62    | 79.74 ± 0.55 | 80.48 ± 0.59 | 82.17 ± 0.47 | 83.06 ± 0.44    |
| CIFAR-100N, noisy fine labels | 58.73 ± 0.71  | 59.42 ± 0.68    | 61.08 ± 0.61 | 61.74 ± 0.64 | 63.26 ± 0.53 | 64.11 ± 0.50    |

Table 5 shows that the largest improvement over conventional cross-entropy was observed in the CIFAR-10N worst-case condition, where quality-aware training improved accuracy by 6.64 percentage points. The proposed method also exceeded DivideMix by 0.89 percentage points in this condition. On CIFAR-100N, the improvement over conventional cross-entropy was 5.38 percentage points.

To complement the numerical comparison in Table 5, Figure 3 visualizes the performance of the evaluated methods under human-generated noisy-label conditions. These conditions provide a more realistic assessment because their annotation errors are associated with ambiguity and visual similarity rather than with purely random label changes.



**Figure 3.** Accuracy comparison under real-world noisy-label conditions

As shown in Figure 3, the proposed method achieved the highest mean accuracy in all four real-world noisy-label conditions. Its improvement over conventional cross-entropy increased from 2.32 percentage points under CIFAR-10N aggregate labels to 6.64 percentage points under the worst-case labels. On CIFAR-100N, the proposed method improved accuracy by 5.38 percentage points. These results indicate that quality-aware weighting remains effective when annotation errors follow realistic and instance-dependent patterns.

The improvements under real-world annotations were slightly smaller than those observed under the highest

synthetic-noise condition. This difference was expected because human annotation errors may be related to semantic ambiguity and class similarity. Such errors are more difficult to identify than randomly corrupted labels because an incorrect human label can remain locally consistent with visually similar samples.

#### Accuracy of Label-Quality Estimation

The effectiveness of quality-aware training depends partly on whether the estimated scores distinguish reliable from unreliable labels. Table 6 summarizes the quality-estimation results.

**Table 6.** Accuracy of Sample-Level Label-Quality Estimation

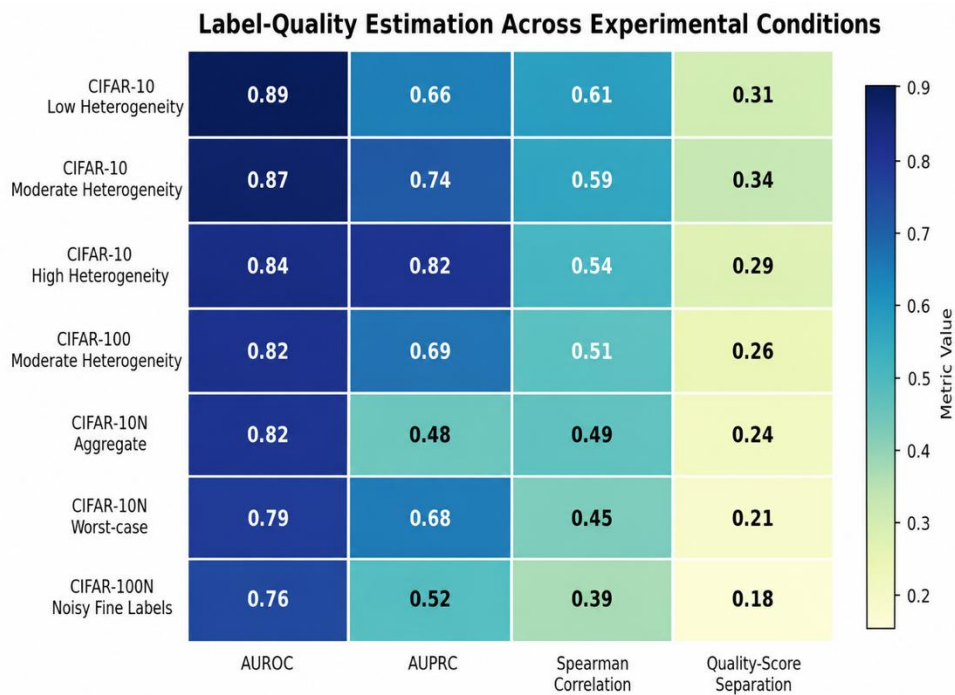
| Condition                         | AUROC | AUPRC | Spearman Correlation | Quality-Score Separation |
|-----------------------------------|-------|-------|----------------------|--------------------------|
| CIFAR-10, low heterogeneity       | 0.89  | 0.66  | 0.61                 | 0.31                     |
| CIFAR-10, moderate heterogeneity  | 0.87  | 0.74  | 0.59                 | 0.34                     |
| CIFAR-10, high heterogeneity      | 0.84  | 0.82  | 0.54                 | 0.29                     |
| CIFAR-100, moderate heterogeneity | 0.82  | 0.69  | 0.51                 | 0.26                     |
| CIFAR-10N, aggregate labels       | 0.82  | 0.48  | 0.49                 | 0.24                     |
| CIFAR-10N, worst-case labels      | 0.79  | 0.68  | 0.45                 | 0.21                     |
| CIFAR-100N, noisy fine labels     | 0.76  | 0.52  | 0.39                 | 0.18                     |

As shown in Table 6, the quality estimator achieved its highest AUROC of 0.89 under low heterogeneous CIFAR-10 noise. Its performance declined as the classification task and noise pattern became more difficult. However, all quality-score separation values remained positive, demonstrating that correctly labeled samples received higher average scores than incorrectly labeled samples.

The higher AUPRC under high synthetic noise was partly influenced by the greater prevalence of corrupted

labels. In contrast, the lower AUPRC for CIFAR-10N aggregate labels reflected the relatively small proportion of incorrect annotations in that condition.

The variation in label-quality estimation across synthetic and real-world conditions is further illustrated in Figure 4. The heatmap jointly presents the discrimination, ranking, and score-separation measures reported in Table 6.



**Figure 4.** Effectiveness of sample-level label-quality estimation across experimental conditions

Figure 4 indicates that the quality estimator produced its strongest discrimination under synthetic CIFAR-10 conditions, with AUROC values ranging from 0.84 to 0.89. Its performance was lower under human-generated noisy labels, particularly for CIFAR-100N, where the AUROC and quality-score separation were 0.76 and 0.18, respectively. Nevertheless, all score-separation

values remained positive, confirming that the estimator consistently assigned higher average quality scores to correct labels than to incorrect labels.

#### Ablation Results

Table 7 presents the ablation results under moderate and high heterogeneous CIFAR-10 noise.

**Table 7.** Ablation Study of the Quality-Estimation Components

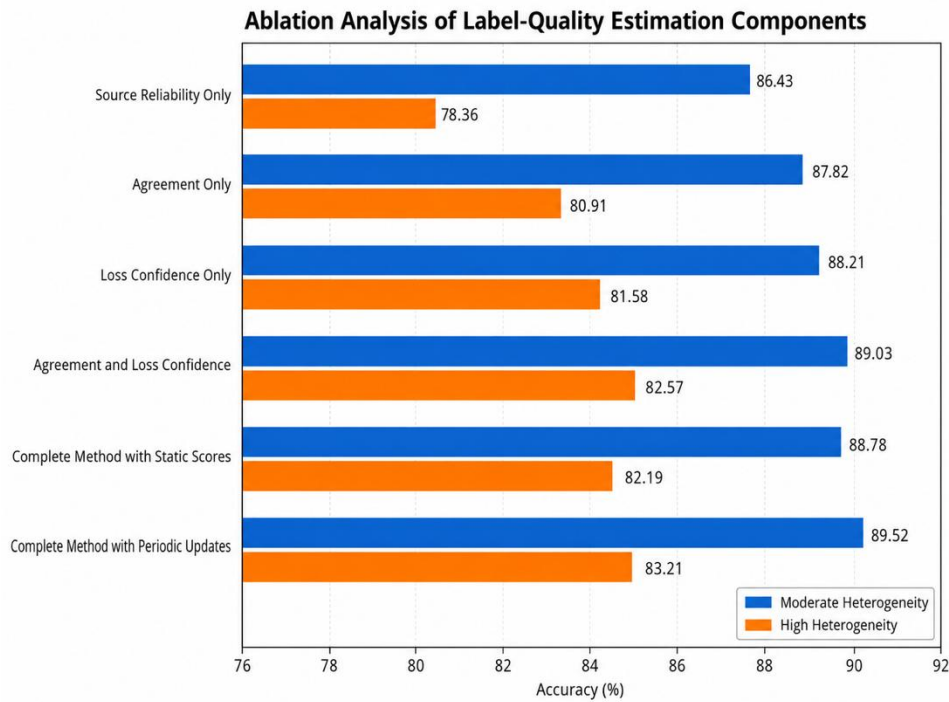
| Quality-Estimation Variant            | Moderate Heterogeneity | High Heterogeneity |
|---------------------------------------|------------------------|--------------------|
| Source reliability only               | 86.43 ± 0.45           | 78.36 ± 0.63       |
| Agreement only                        | 87.82 ± 0.39           | 80.91 ± 0.56       |
| Loss confidence only                  | 88.21 ± 0.37           | 81.58 ± 0.53       |
| Agreement and loss confidence         | 89.03 ± 0.34           | 82.57 ± 0.49       |
| Complete method with static scores    | 88.78 ± 0.36           | 82.19 ± 0.51       |
| Complete method with periodic updates | 89.52 ± 0.31           | 83.21 ± 0.46       |

As summarized in Table 7, source reliability alone produced the weakest result among the quality-aware variants. This finding indicates that source-level information was useful but insufficient for identifying the reliability of individual observations. Loss-based confidence produced slightly better performance than neighbourhood agreement when each component was used independently.

Combining agreement and loss confidence improved accuracy beyond that of either component alone. Adding source reliability and periodically updating the scores

produced the best overall results. Periodic updates increased accuracy by 0.74 percentage points under moderate heterogeneity and 1.02 percentage points under high heterogeneity compared with static quality scores.

To clarify the contribution of each quality-estimation component, Figure 5 compares the ablation variants under both moderate and high levels of heterogeneous noise. The visualization highlights the incremental benefit obtained by combining sample-level and source-level information.



**Figure 5.** Ablation analysis of the label-quality estimation components

Figure 5 demonstrates that source reliability alone provided the lowest accuracy among the quality-aware variants.

#### Sensitivity to Quality-Estimation Errors

Table 8 reports the sensitivity results for CIFAR-10 with moderate heterogeneous noise. The unperturbed quality-aware model achieved an accuracy of 89.52%.

**Table 8.** Sensitivity to Errors in the Estimated Quality Scores

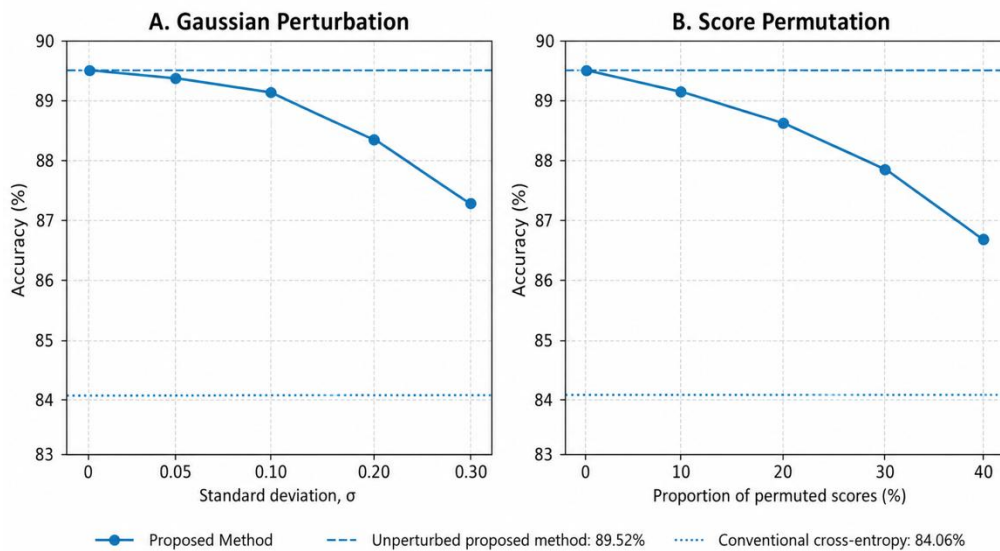
| Perturbation Condition          | Accuracy     | Change From Original |
|---------------------------------|--------------|----------------------|
| No perturbation                 | 89.52 ± 0.31 | –                    |
| Gaussian noise, $\sigma = 0.05$ | 89.41 ± 0.33 | –0.11                |
| Gaussian noise, $\sigma = 0.10$ | 89.18 ± 0.35 | –0.34                |
| Gaussian noise, $\sigma = 0.20$ | 88.42 ± 0.41 | –1.10                |
| Gaussian noise, $\sigma = 0.30$ | 87.31 ± 0.48 | –2.21                |
| Permutation of 10% of scores    | 89.20 ± 0.36 | –0.32                |
| Permutation of 20% of scores    | 88.66 ± 0.40 | –0.86                |
| Permutation of 30% of scores    | 87.84 ± 0.46 | –1.68                |
| Permutation of 40% of scores    | 86.71 ± 0.52 | –2.81                |

Table 8 shows that small perturbations had a limited effect on performance. Gaussian perturbations with standard deviations of 0.05 and 0.10 reduced accuracy by only 0.11 and 0.34 percentage points, respectively. Performance remained above that of the generalized cross-entropy and conventional cross-entropy baselines, even when 30% of the quality scores were permuted.

A larger decline occurred when 40% of the scores were incorrectly reordered. Nevertheless, the resulting accuracy of 86.71% remained 2.65 percentage points higher than conventional cross-entropy. These findings

support the assumption that the proposed method does not require perfectly estimated quality values to provide useful improvements.

The sensitivity of the proposed method to inaccurate quality estimates is visualized in Figure 6. The first panel shows the effect of additive Gaussian perturbations, whereas the second panel presents the effect of progressively permuting the quality-score rankings.



**Figure 6.** Sensitivity of quality-aware training to errors in estimated quality scores

Figure 6 shows a gradual, rather than an abrupt, decline in performance as the quality estimates become less accurate. Gaussian perturbations with standard deviations of 0.05 and 0.10 reduced accuracy by only 0.11 and 0.34 percentage points. Similarly, the model retained an accuracy of 87.84% when 30% of the quality scores were permuted. Even after permuting 40% of the scores, its accuracy remained 2.65 percentage points above conventional cross-entropy. This result supports the robustness of the proposed method to moderate quality-estimation errors.

#### Statistical and Computational Results

The Friedman test identified a significant difference among the six training methods across the matched synthetic and real-world experimental conditions:

$$\chi^2(5) = 47.82, p < 0.001.$$

Holm-adjusted Wilcoxon signed-rank tests showed that the proposed method significantly outperformed conventional cross-entropy ( $p < 0.001$ ), label smoothing ( $p < 0.001$ ), generalized cross-entropy ( $p = 0.002$ ), Co-teaching ( $p = 0.006$ ), and DivideMix ( $p = 0.041$ ). Rank-biserial correlations ranged from 0.48 against DivideMix to 0.98 against conventional cross-entropy, indicating moderate to large effects.

Table 9 summarizes the computational requirements measured on CIFAR-10

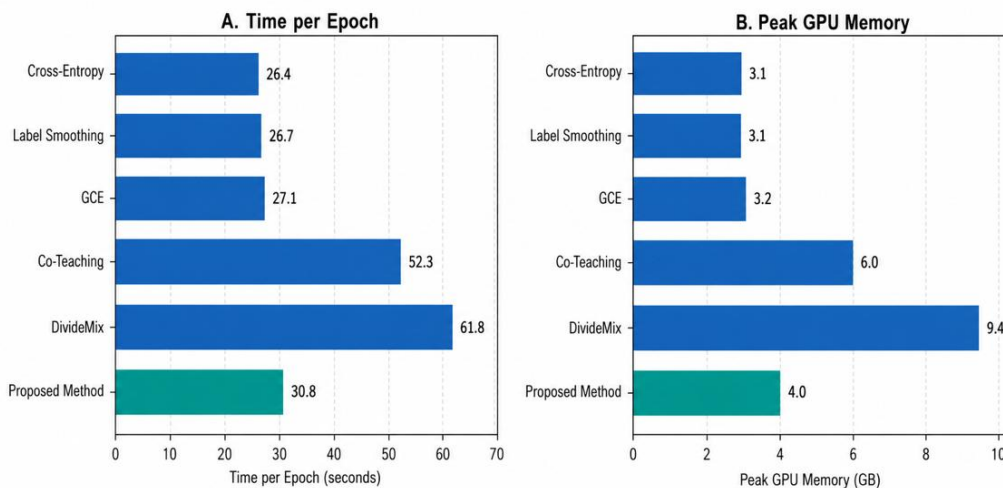
**Table 9.** Computational Requirements of the Compared Methods

| Method                    | Time per Epoch | Peak GPU Memory | Relative Time Overhead |
|---------------------------|----------------|-----------------|------------------------|
| Cross-entropy             | 26.4 s         | 3.1 GB          | —                      |
| Label smoothing           | 26.7 s         | 3.1 GB          | 1.1%                   |
| Generalized cross-entropy | 27.1 s         | 3.2 GB          | 2.7%                   |
| Co-teaching               | 52.3 s         | 6.0 GB          | 98.1%                  |
| DivideMix                 | 61.8 s         | 9.4 GB          | 134.1%                 |
| Proposed method           | 30.8 s         | 4.0 GB          | 16.7%                  |

As shown in Table 9, the proposed method required approximately 16.7% more training time than conventional cross-entropy. Most of this overhead resulted from feature extraction, nearest-neighbour calculation, and periodic quality-score updates. However, its computational cost remained substantially lower than that of Co-teaching and DivideMix, which required training two models simultaneously.

The computational requirements of the compared methods are visualized in Figure 7. Training time and peak GPU memory are presented separately to provide a direct comparison of the additional resources required by each noise-robust strategy.

## Computational Cost of the Compared Training Methods



**Figure 7.** Computational cost of the compared training methods

Figure 7 shows that the proposed method introduced moderate additional computational requirements compared with ordinary cross-entropy. Thus, the proposed method offered a more favorable computational profile while maintaining stronger predictive performance under heterogeneous label noise.

### 3.2 Discussion

The results demonstrate that explicitly accounting for heterogeneous label quality can improve model robustness when training data contain labels of unequal reliability. The proposed method consistently outperformed conventional cross-entropy and other noise-robust approaches as label heterogeneity increased. Its advantage was comparatively small under clean-label conditions but became substantial under moderate and high noise. This pattern indicates that the method did not achieve its improvement through a more general model architecture; instead, the improvement stemmed from its ability to reduce the influence of unreliable annotations.

Conventional cross-entropy treats all observed labels as equally trustworthy. Consequently, samples originating from highly unreliable sources contribute as strongly to parameter updates as samples from reliable sources. This behavior explains the pronounced performance degradation observed under high heterogeneous noise. Label smoothing reduced excessive prediction confidence but did not determine which individual labels were unreliable. It therefore provided only limited protection against strongly heterogeneous errors.

Generalized cross-entropy was more robust than ordinary cross-entropy because it reduced the influence of high-loss observations. However, sample loss alone cannot always distinguish corrupted labels from difficult but correctly labeled cases. The proposed method addressed this limitation by combining loss-based confidence with neighbourhood agreement and source-level reliability. A sample received a low score only

when several indicators suggested that its observed label was unreliable.

Co-teaching and DivideMix achieved stronger results than loss-based baselines, particularly under severe noise. Nevertheless, both methods largely depend on separating samples into likely clean and noisy groups. Such binary or mixture-based separation may be restrictive when label reliability varies continuously. In contrast, the proposed approach retained every observation while adjusting its influence through a continuous weight. A difficult sample could therefore remain involved in training without contributing as strongly as a highly reliable sample.

The ablation results confirm that no single quality indicator was sufficient. Source reliability captured systematic differences among annotation providers but could not represent within-source variation. A generally reliable source may still produce individual mistakes, while a weak source may occasionally provide correct annotations. Similarly, loss of confidence was informative, but could assign low scores to difficult clean samples. Combining loss evidence with annotation or neighbourhood agreement reduced these weaknesses.

Periodic quality-score updates also improved performance. The representations produced immediately after warm-up training were informative but remained incomplete. As training progressed, the feature space became more class-discriminative, allowing neighbourhood agreement and loss confidence to be estimated more accurately. Updating the scores every five epochs provided a practical balance between adaptation and computational cost.

The real-world experiments provide important evidence because human annotation errors do not follow simple random-noise assumptions. The proposed method achieved lower quality-detection AUROC on CIFAR-10N and CIFAR-100N than under synthetic noise, indicating that real annotation errors were more difficult to identify. Despite this decline, the model continued to

improve its predictive performance. This finding suggests that moderately informative quality estimates can be sufficient when used as soft weights rather than as absolute decisions to delete samples.

The sensitivity results further support this interpretation. Small and moderate perturbations to the quality scores caused only limited performance degradation. The method remained superior to conventional training even when a substantial proportion of quality scores were incorrectly ordered. Therefore, the third hypothesis was supported: quality-aware training remained useful under moderate quality-estimation errors.

### 3.2.1 Implications

The findings support the three research hypotheses. First, quality-aware sample weighting improved predictive performance compared with conventional unweighted training. Second, the proposed method maintained greater robustness as label noise and source heterogeneity increased. Third, the method remained effective under moderate errors in quality estimation. Small perturbations to the estimated scores caused only limited performance degradation, indicating that perfectly accurate label-quality estimates were not required to obtain meaningful benefits.

The ablation analysis showed that agreement, loss-based confidence, and source reliability provided complementary information. Source reliability alone was insufficient because samples from the same source could still vary in correctness. Combining sample-level indicators with source-level information and periodically updating the quality scores produced the strongest performance. The method also introduced substantially less computational overhead than approaches requiring two simultaneously trained networks.

### 3.2.2 Research contribution

This study makes four principal contributions.

First, it formulates heterogeneous label quality as a sample- and source-dependent reliability problem rather than representing noise only through a single global corruption rate.

Second, it introduces a unified quality-estimation mechanism that combines annotation or neighbourhood agreement, loss-based confidence, and source reliability. The combination reduces dependence on any single potentially misleading indicator.

Third, the proposed method integrates the estimated scores into a continuous weighted loss. Unlike sample-removal and binary clean-noisy separation strategies, this mechanism preserves uncertain observations while controlling their influence.

Fourth, the study evaluates robustness not only across synthetic and human-generated noise conditions but also under deliberately perturbed quality estimates. The sensitivity results demonstrate that perfect knowledge of label correctness is not required to obtain performance benefits.

Collectively, these contributions provide an empirically supported and computationally feasible strategy for learning from datasets with unequally reliable labels.

### 3.2.3 Limitations

Several limitations should be considered when interpreting the findings. First, the experiments were restricted to image-classification benchmarks. The effectiveness of the method has not yet been established for text classification, tabular prediction, regression, object detection, or multilabel learning.

Second, the synthetic annotation sources were constructed using predefined corruption probabilities. Although this procedure enabled controlled experiments, real annotation sources may exhibit more complex behavior, including class-dependent expertise, correlated mistakes, and changes in reliability over time.

Finally, although five independent runs were sufficient to estimate the main variation in the benchmark experiments, evaluation across more architectures, datasets, and random initializations would provide stronger evidence of generalizability.

### 3.2.4 Suggestions

Future research should evaluate the proposed approach on additional data modalities and learning tasks. Text classification, medical records, financial risk prediction, and cybersecurity datasets would be particularly relevant, as labels in these domains are often produced by multiple experts or automated rules.

Future weighting methods could incorporate uncertainty estimation, prediction consistency across augmentations, or ensemble disagreement into the quality score. These indicators may help distinguish difficult clean samples from genuinely mislabeled observations.

## 4. CONCLUSION

This study proposed a quality-aware training method for learning from datasets with heterogeneous label reliability. Unlike conventional training, which treats all observed labels as equally trustworthy, the proposed approach estimated a continuous quality score for each sample by combining annotation or neighbourhood agreement, loss-based confidence, and source-level reliability. These scores were then incorporated into a weighted cross-entropy objective so that reliable labels contributed more strongly to model optimization, while uncertain labels retained a smaller but nonzero influence.

The experimental results demonstrated that the proposed method consistently improved predictive performance under synthetic and real-world noisy-label conditions. The improvements became more substantial as label-quality heterogeneity increased. Under high heterogeneous noise, the method achieved accuracies of 83.21% on CIFAR-10 and 54.62% on CIFAR-100, compared with 73.48% and 45.92%, respectively, for

conventional cross-entropy. The method also produced the highest accuracy across the CIFAR-10N and CIFAR-100N conditions, including 83.06% under the CIFAR-10N worst-case labels and 64.11% under CIFAR-100N noisy fine labels.

Overall, the study demonstrates that label reliability should be treated as a continuous and sample-dependent property rather than as a uniform dataset characteristic or a binary clean-noisy distinction. The proposed method provides a practical solution for crowdsourced, weakly supervised, and multi-source learning environments in which completely verified annotations are unavailable or expensive to obtain. Future research should evaluate the framework on other data modalities and learning tasks, incorporate class-dependent and time-varying source reliability, and investigate fairness mechanisms to prevent difficult but valid minority samples from being systematically underweighted.

## 5. ACKNOWLEDGEMENT

This work benefited greatly from numerous inspiring discussions with colleagues and peers who generously shared their perspectives on robust learning and data quality. We are also indebted to the anonymous reviewers, whose sharp critiques and constructive suggestions pushed us to refine both our theoretical framing and empirical validation.

## 6. AUTHOR CONTRIBUTION STATEMENT

The research concept and technique were devised and designed by SW, HMK, and AAS. SW and HMK created the software implementation, carried out the real-world and synthetic experiments, and gathered the computational outcomes. The statistical analyzes and label-quality assessment were analyzed with assistance from AAS and SW. The text was drafted by SW, HMK, and AAS, who also assessed the experimental results and made final revisions for publishing.

## AUTHOR INFORMATION

### Corresponding Authors

Wu Shukun, Guangxi Science & Technology Normal University, China

 <https://orcid.org/0000-0003-0371-4167>

Email: [wuskctrl@outlook.com](mailto:wuskctrl@outlook.com)

### Authors

Hafiz Muhammad Kurniawan, INTI International University, Malaysia

 <https://orcid.org/0009-0007-5069-0811>

Email: [kurni.hafiz2002@gmail.com](mailto:kurni.hafiz2002@gmail.com)

Adolf Asih Suprianto, Politeknik Enjinering Indorama, Indonesia

 <https://orcid.org/0009-0009-7945-8754>

Email: [adolf@pei.ac.id](mailto:adolf@pei.ac.id)

## REFERENCE

- Bernhardt, M., Castro, D. C., Tanno, R., Schwaighofer, A., Tezcan, K. C., Monteiro, M., Bannur, S., Lungren, M. P., Nori, A., Glocker, B., Alvarez-Valle, J., & Oktay, O. (2022). Active label cleaning for improved dataset quality under resource constraints. *Nature Communications*, 13(1). <https://doi.org/10.1038/s41467-022-28818-3>
- Braun, D. (2024). I beg to differ: how disagreement is handled in the annotation of legal machine learning data sets. *Artificial Intelligence and Law*, 32(3), 839–862. <https://doi.org/10.1007/s10506-023-09369-4>
- Brigato, L., Barz, B., Iocchi, L., & Denzler, J. (2022). Image Classification With Small Datasets: Overview and Benchmark. *IEEE Access*, 10, 49233–49250. <https://doi.org/10.1109/ACCESS.2022.3172939>
- Capstick, A., Palermo, F., Cui, T., & Barnaghi, P. (2025). Training neural networks on data sources with unknown reliability. *Information Fusion*, 123, 103239. <https://doi.org/10.1016/J.INFFUS.2025.103239>
- Chase, R. J., Harrison, D. R., Burke, A., Lackmann, G. M., & McGovern, A. (2022). A Machine Learning Tutorial for Operational Meteorology. Part I: Traditional Machine Learning. *Weather and Forecasting*, 37(8), 1509–1529. <https://doi.org/10.1175/WAF-D-22-0070.1>
- Chen, Y., Xu, K., Zhou, P., Ban, X., & He, D. (2022). Improved cross entropy loss for noisy labels in vision leaf disease classification. *IET Image Processing*, 16(6), 1511–1519. <https://doi.org/10.1049/ipr2.12402>
- de Vos, B. D., Jansen, G. E., & Išgum, I. (2023). Stochastic co-teaching for training neural networks with unknown levels of label noise. *Scientific Reports* 2023 13:1, 13(1), 16875-. <https://doi.org/10.1038/s41598-023-43864-7>
- Gao, Y., Fu, J., Wang, Y., & Guo, Y. (2024). Typicality- and instance-dependent label noise-combating : a novel framework for simulating and combating real-world noisy labels for endoscopic polyp classification. *Visual Computing for Industry, Biomedicine, and Art*, 7. <https://doi.org/10.1186/s42492-024-00162-x>
- Gil-González, J., Daza-Santacoloma, G., Cárdenas-Peña, D., Orozco-Gutiérrez, A., & Álvarez-Meza, A. (2025). Generalized cross-entropy for learning from crowds based on correlated chained Gaussian processes. *Results in Engineering*, 25(January), 103863. <https://doi.org/10.1016/j.rineng.2024.103863>
- González-Santoyo, C., Renza, D., & Moya-Albor, E. (2025). Identifying and Mitigating Label Noise in Deep Learning for Image Classification. *Technologies* 2025, Vol. 13, Page 132, 13(4), 132. <https://doi.org/10.3390/TECHNOLOGIES13040132>

- Hafiz, A. M., Bhat, R. A., & Hassaballah, M. (2022). Image classification using convolutional neural network tree ensembles. *Multimedia Tools and Applications* 2022 82:5, 82(5), 6867–6884. <https://doi.org/10.1007/S11042-022-13604-6>
- Haliburton, L., Leusmann, J., Welsch, R., Ghebremedhin, S., Isaakidis, P., Schmidt, A., & Mayer, S. (2025). Uncovering labeler bias in machine learning annotation tasks. *AI and Ethics*, 5(3), 2515–2528. <https://doi.org/10.1007/s43681-024-00572-w>
- Han, B., Sun, Y. X., Zhang, Y. L., Zhang, L., Hu, H., Li, L., Zhou, J., Ye, G., & He, H. (2024). Collaborative Refining for Learning from Inaccurate Labels. *Advances in Neural Information Processing Systems*, 37, 92745–92768. <https://doi.org/10.52202/079017-2945>
- Herde, M., Huseljic, D., Sick, B., & Calma, A. (2021). A Survey on Cost Types, Interaction Schemes, and Annotator Performance Models in Selection Algorithms for Active Learning in Classification. *IEEE Access*, 9, 166970–166989. <https://doi.org/10.1109/ACCESS.2021.3135514>
- Higashimoto, R., Yoshida, S., & Muneyasu, M. (2024). CRAS: Curriculum Regularization and Adaptive Semi-Supervised Learning with Noisy Labels. *Applied Sciences (Switzerland)*, 14(3). <https://doi.org/10.3390/app14031208>
- Klie, J. C., de Castilho, R. E., & Gurevych, I. (2024). Analyzing Dataset Annotation Quality Management in the Wild. *Computational Linguistics*, 50(3), 817–866. [https://doi.org/10.1162/COLI\\_A\\_00516](https://doi.org/10.1162/COLI_A_00516)
- Klie, J. C., Webber, B., & Gurevych, I. (2023). Annotation Error Detection: Analyzing the Past and Present for a More Coherent Future. *Computational Linguistics*, 49(1), 157–198. [https://doi.org/10.1162/coli\\_a\\_00464](https://doi.org/10.1162/coli_a_00464)
- Kukreja, V., Abraham, A., Kalaiselvi, K., Thilak, K. D., Hariharan, S., & Chen, S. Y. (2023). Machine Learning for Data Fusion: A Fuzzy AHP Approach for Open Issues. *Computers, Materials and Continua*, 77(3), 2899–2914. <https://doi.org/10.32604/cmc.2023.045136>
- Lazaros, K., Vrahatis, A. G., & Kotsiantis, S. (2026). Human-in-the-Loop Artificial Intelligence: A Systematic Review of Concepts, Methods, and Applications. *Entropy*, 28(4), 1–50. <https://doi.org/10.3390/e28040377>
- Liu, Z., & Letchmunan, S. (2024). Representing uncertainty and imprecision in machine learning: A survey on belief functions. *Journal of King Saud University - Computer and Information Sciences*, 36(1). <https://doi.org/10.1016/j.jksuci.2023.101904>
- Liu, Z., Qian, P., Wang, X., Zhuang, Y., Qiu, L., & Wang, X. (2023). Combining Graph Neural Networks with Expert Knowledge for Smart Contract Vulnerability Detection. *IEEE Transactions on Knowledge and Data Engineering*, 35(2), 1296–1310. <https://doi.org/10.1109/TKDE.2021.3095196>
- Mohammed, S., Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., Naumann, F., & Harmouch, H. (2025). The effects of data quality on machine learning performance on tabular data. *Information Systems*, 132(March), 102549. <https://doi.org/10.1016/j.is.2025.102549>
- Northcutt, C. G., Athalye, A., & Mueller, J. (2021). Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. *Advances in Neural Information Processing Systems*. <https://arxiv.org/pdf/2103.14749>
- Northcutt, C. G., Jiang, L., & Chuang, I. L. (2021). Confident Learning: Estimating Uncertainty in Dataset Labels. *Journal of Artificial Intelligence Research*, 70, 1373–1411. <https://doi.org/10.1613/JAIR.1.12125>
- Özdemir, Z., Keles, H. Y., & Tanrıöver, Ö. Ö. (2026). CLoE: Curriculum Learning on Endoscopic Images for Robust MES Classification. *IEEE Access*, 14, 30441–30454. <https://doi.org/10.1109/ACCESS.2026.3666962>
- Pradhan, V. K., Schaekermann, M., & Lease, M. (2022). In Search of Ambiguity: A Three-Stage Workflow Design to Clarify Annotation Guidelines for Crowd Workers. *Frontiers in Artificial Intelligence*, 5(May), 1–16. <https://doi.org/10.3389/frai.2022.828187>
- Priestley, M., O'Donnell, F., & Simperl, E. (2023). A Survey of Data Quality Requirements That Matter in ML Development Pipelines. *Journal of Data and Information Quality*, 15(2). <https://doi.org/10.1145/3592616>
- Rashidi, H. H., Tran, N., Albahra, S., & Dang, L. T. (2021). Machine learning in health care and laboratory medicine: General overview of supervised learning and Auto-ML. *International Journal of Laboratory Hematology*, 43(S1), 15–22. <https://doi.org/10.1111/ijlh.13537>
- Ratner, A., Bach, S. H., Ehrenberg, H., Fries, J., Wu, S., & Ré, C. (2019). Snorkel: rapid training data creation with weak supervision. *The VLDB Journal* 2019 29:2, 29(2), 709–730. <https://doi.org/10.1007/S00778-019-00552-1>
- Shetty, S. H., Shetty, S., Singh, C., & Rao, A. (2022). Supervised machine learning: Algorithms and applications. *Fundamentals and Methods of Machine and Deep Learning: Algorithms, Tools, and Applications*, 1–16. <https://doi.org/10.1002/9781119821908.CH1;CTYPE:STRING:BOOK>
- Song, B., Zhao, S., Dang, L., Wang, H., & Xu, L. (2025). A survey on learning from data with label noise via deep neural networks. *Systems Science & Control Engineering*, 13(1), 2488120. <https://doi.org/10.1080/21642583.2025.2488120>
- Wang, Q., Han, B., Liu, T., Niu, G., Yang, J., & Gong, C. (2021). Tackling Instance-Dependent Label Noise via a Universal Probabilistic Model. *35th AAAI Conference on Artificial Intelligence, AAAI 2021*, 11B, 10183–10191. <https://doi.org/10.1609/aaai.v35i11.17221>

- Wang, Y., Han, X., Li, C., Luo, L., Yin, Q., Zhang, J., Peng, G., Shi, D., & He, M. (2024). Impact of Gold-Standard Label Errors on Evaluating Performance of Deep Learning Models in Diabetic Retinopathy Screening: Nationwide Real-World Validation Study. *Journal of Medical Internet Research*, 26, 1–12. <https://doi.org/10.2196/52506>
- Xu, C., Song, P., Tang, S., Guo, D., & Yang, X. (2025). Alleviating Confirmation Bias in Learning with Noisy Labels via Two-Network Collaboration. *ACM Transactions on Intelligent Systems and Technology*, 16(4). <https://doi.org/10.1145/3723009>
- Yang, Y., Zhao, Z. Q., Wu, G., Zhuo, X., Liu, Q., Bai, Q., & Li, W. (2024). A Lightweight, Effective, and Efficient Model for Label Aggregation in Crowdsourcing. *ACM Transactions on Knowledge Discovery from Data*, 18(4). <https://doi.org/10.1145/3630102>
- Ye, C., Duan, H., Zhang, H., Zhang, H., Wang, H., & Dai, G. (2023a). Multi-Source Data Repairing: A Comprehensive Survey. *Mathematics*, 11(10), 1–22. <https://doi.org/10.3390/math11102314>
- Ye, C., Duan, H., Zhang, H., Zhang, H., Wang, H., & Dai, G. (2023b). Multi-Source Data Repairing: A Comprehensive Survey. *Mathematics*, 11(10), 1–22. <https://doi.org/10.3390/math11102314>
- Zhang, T., Yi, C., Song, Y., Wei, B., & Chen, Z. (2026). Hyperbolic DivideMix: Leveraging Poincaré Ball Embeddings for Robust Human Activity Recognition under Label Noise. *IEEE Sensors Journal*. <https://doi.org/10.1109/JSEN.2026.3680607>
- Zhang, Z., & Sabuncu, M. R. (2018). Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. *Advances in Neural Information Processing Systems, 2018-December*, 8778–8788. <https://arxiv.org/pdf/1805.07836>
- Zhao, X., Huang, P., & Shu, X. (2022). Wavelet-Attention CNN for image classification. *Multimedia Systems 2022* 28:3, 28(3), 915–924. <https://doi.org/10.1007/S00530-022-00889-8>
- Zhong, Y., Du, B., & Xu, C. (2021). Learning to reweight examples in multi-label classification. *Neural Networks*, 142, 428–436. <https://doi.org/10.1016/J.NEUNET.2021.03.022>