LOCAL LOOK-AHEAD ENSEMBLING: *Do We Really Need a Second Network for Learning with Noisy Labels?*

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent learning with noisy labels (LNL) methods are commonly formulated as semi-supervised learning problems, where a teacher is used for sample selection and pseudo-label generation. Among existing approaches, co-training-based methods have achieved strong performance by maintaining two networks that exchange information during training. However, this design nearly doubles the computational cost, raising the question of whether a second network is truly necessary. In this paper, we introduce Local Look-Ahead Ensembling (LLE), a teacher construction strategy designed to replace the peer network in co-training-based LNL methods. Instead of training a second model from scratch, LLE constructs several local models by performing a small number of look-ahead optimization steps from the current model and aggregates their predictions to form a teacher. As a result, existing two-network LNL frameworks can be converted into single-network methods without changing their overall training logic. We first analyze LLE in a controlled semi-supervised learning setting and show that it primarily modifies pseudo-labels on low-confidence samples while providing substantially more accurate predictions on these examples. We then integrate LLE into representative co-training-based LNL methods, including DivideMix, UniCon, and C2D. Experimental results on CIFAR-10 and CIFAR-100 demonstrate that LLE preserves performance at low and moderate noise rates, consistently improves accuracy under severe label noise, and significantly reduces training cost by eliminating the need for a second network.

1 INTRODUCTION

Deep learning models have achieved remarkable success across a wide range of applications. However, their performance heavily relies on the availability of large-scale labeled datasets, whose annotations are often imperfect in practice. Since deep neural networks can eventually memorize incorrect labels Zhang et al. (2016), training on noisy datasets often leads to degraded generalization performance.

Many modern learning with noisy labels (LNL) methods address this challenge by viewing LNL as a semi-supervised learning (SSL) problem. Under this perspective, the training data are divided into two groups at each epoch: samples whose labels are estimated to be correct and samples suspected to be noisy. The first group is used for supervised learning, while the second group participates through an SSL objective. Consequently, two components play a central role in most LNL methods: sample selection and pseudo-label generation.

A straightforward approach for generating pseudo-labels is to use the model's own predictions. However, such self-training strategies are prone to confirmation bias Arazo et al. (2020), where prediction errors are repeatedly reinforced during training. To alleviate this issue, many successful LNL methods adopt a co-training paradigm Qiao et al. (2018); Ke et al. (2019). In this setting, two networks are trained simultaneously and exchange information for sample selection and pseudo-label generation. The diversity between the two networks helps reduce confirmation bias and has become a key ingredient of state-of-the-art methods such as DivideMix Li et al. (2020), UniCon Karim et al. (2022), C2D Zheltonozhskii et al. (2022), Co-teaching Han et al. (2018), and ELR Liu et al. (2020).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

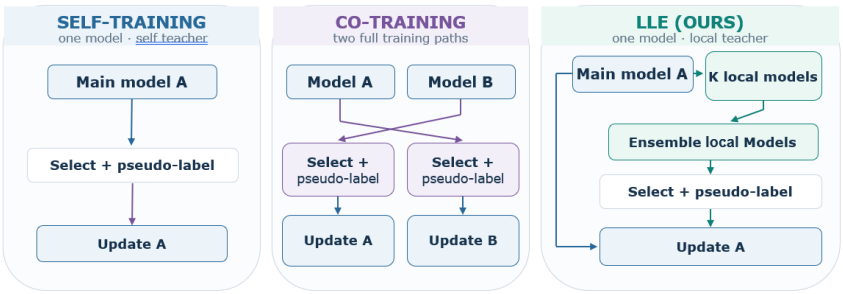


Figure 1: Comparison of self-training, co-training, and LLE-training(Ours) in the model training pipeline for the LNL problem. **Left:** the self-training approach, where in each epoch the model itself is used for both sample selection and label correction. **Middle:** the co-training approach, where two independent networks are trained in parallel and, at each epoch, each network relies on its peer for sample selection and label correction. **Right:** our proposed method, which uses a single network; in each epoch, several local models are first constructed from the main model, and the ensemble of these models is then used for sample selection and label correction.

Despite its effectiveness, co-training introduces a substantial computational burden. Since both networks follow essentially the same training pipeline, the cost of training is nearly doubled. This raises a natural question: *Do we really need a second network in co-training-based LNL?*

In this paper, we investigate whether the role of the peer network can be replaced without maintaining an entirely separate learning trajectory. Our starting observation is that useful predictive diversity may already exist in the local neighborhood of a single model. If this diversity can be extracted effectively, then the information traditionally provided by the second network may not require training a second model from scratch.

Based on this idea, we propose **Local Look-Ahead Ensembling (LLE)**. At each epoch, LLE constructs several local models by performing a small number of look-ahead optimization steps from the current model using randomly sampled mini-batches. The predictions of these local models are then aggregated to form a teacher, which replaces the peer network in co-training-based LNL methods. As illustrated in Figure 1, LLE preserves the overall training logic of existing algorithms while eliminating the need for a second semi-supervised learner.

To better understand the behavior of the proposed teacher, we first analyze LLE in a controlled SSL setting. We show that LLE modifies the pseudo-labels of only a small subset of samples and that these modifications occur primarily on low-confidence examples. Moreover, the injected labels are substantially more accurate than those produced by the main model, allowing useful information from difficult samples to enter the training process.

We evaluate LLE by replacing the second network in several representative co-training-based LNL methods, including DivideMix, UniCon, and C2D. Experimental results show that LLE preserves performance at low and moderate noise rates while consistently improving accuracy under severe label noise. At the same time, the computational cost is significantly reduced because training a complete second network is no longer required.

Our Contributions.

- We revisit a fundamental design choice in modern learning with noisy labels methods: the use of a second network in co-training frameworks. We investigate whether the information provided by a peer network can be recovered without maintaining a separate training trajectory.
- We propose **Local Look-Ahead Ensembling (LLE)**, a plug-and-play teacher construction strategy that replaces the second network in co-training-based LNL methods. LLE builds a teacher by ensembling multiple local models obtained through short look-ahead optimization steps from the current model.

- We provide an empirical analysis of LLE in a controlled semi-supervised learning setting. Our analysis shows that LLE primarily modifies predictions on low-confidence samples and injects substantially more accurate information on these difficult examples than the main model.
- We demonstrate that LLE can effectively replace the second network in representative co-training-based LNL methods, including DivideMix, UniCon, and C2D. Experimental results show that LLE preserves performance at low and moderate noise rates, consistently improves accuracy under severe label noise, and significantly reduces computational cost.

2 Related Works

A wide variety of approaches have been proposed to address learning with noisy labels (LNL). One line of work focuses on label correction, where the objective is to progressively replace noisy annotations with cleaner targets during training. Representative examples include Bootstrapping Reed et al. (2014), SELC Lu & He (2022), and ProSelfLC Wang et al. (2021). These methods iteratively refine labels using model predictions and subsequently retrain the model on the corrected labels.

Another popular direction relies on estimating the noise transition matrix Patrini et al. (2017); Xia et al. (2019). Under this framework, the relationship between clean and noisy labels is explicitly modeled, allowing the training objective to be corrected by accounting for the estimated corruption process.

Sample selection methods represent another major category of LNL algorithms. Methods such as Co-teaching Han et al. (2018) and JoCoR Wei et al. (2020) attempt to identify samples with reliable labels during training and update the model primarily using these selected examples. The underlying motivation is based on the observation that deep networks tend to fit clean samples before memorizing noisy ones.

Several studies have also explored early-learning and early-stopping phenomena in deep networks. Representative methods include Bai et al. (2021); Song et al. (2019); Yuan et al. (2024), which seek to exploit the early stages of training before the network begins to overfit noisy labels. By leveraging this behavior, these approaches aim to prevent the memorization of corrupted annotations.

A complementary line of research focuses on designing robust loss functions that are less sensitive to label noise than the standard cross-entropy loss. Examples include Ghosh et al. (2017); Zhang & Sabuncu (2018), where alternative objectives are proposed to reduce the influence of incorrectly labeled samples during optimization.

More recently, the most successful LNL methods have largely adopted a semi-supervised learning perspective. Under this formulation, samples estimated to be clean are used for supervised learning, while the remaining samples participate through pseudo-labeling and consistency-based objectives. Representative state-of-the-art methods include DivideMix Li et al. (2020), ELR Liu et al. (2020), UniCon Karim et al. (2022), and C2D Zheltonozhskii et al. (2022). Within this paradigm, the quality of pseudo-label generation becomes a critical factor, motivating the use of teacher-based and co-training strategies to alleviate confirmation bias.

3 Preliminaries

3.1 Learning with Noisy Labels as Semi-Supervised Learning

Let

$$\mathcal{D} = \{(x_i, \tilde{y}_i)\}_{i=1}^{N}$$

denote a training dataset with noisy labels, where $\tilde{y}_i$ may differ from the unknown clean label y_i. Modern LNL methods commonly reformulate this problem as a semi-supervised learning task. At each epoch, samples are partitioned into two subsets: a clean subset $\mathcal{D}_c$ containing samples whose labels are believed to be correct, and a noisy subset $\mathcal{D}_n$. The clean subset is used for supervised learning, while the remaining samples participate through an SSL objective using pseudo-labels

generated by a teacher. Consequently, the quality of sample selection and pseudo-label generation plays a central role in the performance of LNL algorithms.

3.2 Co-Training in LNL

Many state-of-the-art LNL methods employ two networks, f_1 and f_2, trained simultaneously on the same noisy dataset. During training, each network uses information provided by its peer for sample selection, pseudo-label generation, or both. For example, in methods such as DivideMix, UniCon, and C2D, the predictions of one network are used to construct training targets for the other network. The resulting interaction helps reduce confirmation bias and has proven highly effective in practice. However, maintaining a second network substantially increases computational cost, since the entire training pipeline must be executed twice. This raises the following question:

> Can the information provided by the peer network be recovered without training a second model?

The proposed Local Look-Ahead Ensembling is designed to address this question.

4 Proposed Method

In this section, we introduce Local Look-Ahead Ensembling (LLE), describe how it can replace the peer network in co-training-based LNL methods, and discuss its connection to recent observations on diversity and local exploration in parameter space.

4.1 Local Look-Ahead Ensembling

The central idea of LLE is to construct a teacher from the local neighborhood of the current model. Instead of training a second network from scratch, LLE generates several local models by performing short look-ahead optimization trajectories from the current model and aggregates their predictions into a single teacher.

Let w_t denote the parameters of the main model at the beginning of epoch t, and let $\mathcal{D}_{lle}$ denote a set of clean labeled samples used to construct local models. At each epoch, LLE generates K local models. For each local model k, we first sample s mini-batches from $\mathcal{D}_{lle}$:

$$B_{k,1}^{(t)}, B_{k,2}^{(t)}, \ldots, B_{k,s}^{(t)} \sim \mathcal{D}_{lle}.$$

The local model is initialized from the current model,

$$\hat{w}_0^{(k)} = w_t,$$

and then updated for s look-ahead optimization steps:

$$\hat{w}_{u+1}^{(k)} = U\left(\hat{w}_u^{(k)}, B_{k,u+1}^{(t)}\right), \qquad u = 0, \ldots, s-1, \tag{1}$$

where $U(\cdot)$ denotes a single optimization step. The resulting local model is defined as

$$w_t^{(k)} := \hat{w}_s^{(k)}.$$

After constructing all local models, LLE forms a teacher by averaging their predictive distributions. For an input sample x, the teacher prediction is given by

$$T_t(x) = \frac{1}{K} \sum_{k=1}^{K} f(x; w_t^{(k)}),$$

where $f(x; w_t^{(k)})$ denotes the probabilistic output of the k-th local model. Whenever a hard pseudo-label is required, the teacher prediction is converted into

$$\hat{y}_t(x) = \arg\max_{c \in \mathcal{C}} T_t(x)[c],$$

where $\mathcal{C}$ denotes the set of classes.

In Eq. 1, the local models use exactly the same optimizer and optimization hyperparameters as the main model, including the learning rate, momentum, and weight decay. Furthermore, the mini-batch size used for constructing local models is identical to that used during the training of the main model. Algorithm 1 summarizes the procedure used to construct the LLE teacher at epoch t.

Algorithm 1: Local Look-Ahead Ensembling (LLE) Teacher at Epoch t

Input: Model weights w_t, clean set $\mathcal{D}_{lle}$, number of local models K, look-ahead steps s,
Output: Teacher $T_t(\cdot)$

for $k = 1$ **to** K **do**
  $w_0^{(k)} \leftarrow w_t$; `// Initialize local model from main model`
  for $u = 1$ **to** s **do**
    Sample a mini-batch $B_{k,u}^{(t)}$ from $\mathcal{D}_{lle}^{(t)}$;
    $w_u^{(k)} \leftarrow U(w_u^{(k)}, B_{k,u}^{(t)})$; `// One look-ahead update`
  $w_t^{(k)} \leftarrow w_s^{(k)}$; `// Final local model`
;
`/* Teacher: */`
for *each input* x **do**
  $T_t(x) \leftarrow \frac{1}{K}\sum_{k=1}^{K} f(x; w_t^{(k)})$; `// Ensemble prediction`
return $T_t(\cdot)$

4.2 LLE in Learning with Noisy Labels

Most modern learning with noisy labels (LNL) methods adopt a semi-supervised learning framework. Within this framework, each training epoch typically consists of two key components: sample selection and pseudo-label generation. To reduce confirmation bias, many state-of-the-art approaches rely on a co-training paradigm, where two networks are trained simultaneously and exchange information during training. One network uses the predictions of the other network for sample selection, pseudo-label generation, or both.

The central goal of LLE is not to modify the overall training procedure of existing LNL algorithms. Instead, LLE is designed to replace the peer network used in co-training. Given a co-training-based method such as DivideMix, UniCon, or C2D, we retain all components of the original algorithm and replace only the information provided by the second network. More specifically, at epoch t, we first construct the LLE teacher described in the previous subsection. The teacher predictions are then used wherever the original algorithm requires information from the peer network. As a result, the original two-network training pipeline is converted into a single-network framework without changing the sample-selection or pseudo-labeling mechanisms of the underlying method. Figure 1 illustrates this replacement process.

For evaluation, we report the accuracy of the ensemble consisting of the main model and all local models. This is analogous to the evaluation protocol commonly adopted in co-training methods, where the final prediction is obtained from multiple models rather than from a single network.

A practical consideration in LLE is the construction of the dataset $\mathcal{D}_{lle}$ used for generating local models. The quality of the resulting teacher directly depends on the quality of this dataset. A natural choice is to use the samples identified as clean by the sample-selection module in the previous epoch. However, under severe label noise, this set may contain a substantial number of incorrectly selected samples, which can negatively affect the local models and consequently degrade the quality of the teacher. Therefore, in our experiments involving very high noise rates, we assume access to a small clean validation set and use this set to construct the local models. The effect of this design choice is investigated in the ablation studies presented in Section 5.

4.3 Relation To Other Area

As discussed above, in LLE, the accuracy of the ensemble of the main model and its local models—obtained through local exploration around the main model—is reported as the final accuracy. In our experiments, we observe that LLE achieves performance comparable to, and in some cases even better than, co-training-based methods, despite the fact that the reported accuracy of co-training methods is obtained by ensembling two separately trained networks.

This observation suggests that, in LNL, using two networks and two separate learning trajectories does not necessarily provide sufficient functional diversity to yield a substantial ensemble benefit.

In contrast, LLE demonstrates that comparable or even better performance can be achieved through local exploration around a single model.

This finding is also consistent with the observations in Lion et al. (2024). The authors show that models located in different basins can provide greater functional diversity than models constrained to a single basin, and that increasing connectivity is generally accompanied by reduced predictive diversity and lower ensemble performance. At the same time, they demonstrate that the diversity associated with multiple basins can, to a large extent, be rediscovered within a single basin, particularly when information from a model outside the basin is available to guide the search. Without such information from other basins, however, rediscovering this diversity remains challenging.

Within this context, our results can be interpreted as follows. While independent training on clean data can produce models with high functional diversity, the interaction between networks in co-training for LNL may not provide the same degree of independence. The use of similar pseudo-labels, overlapping sample selection, and a shared noisy dataset may couple the optimization trajectories of the two networks, thereby reducing their effective diversity. Consequently, differences in parameter space do not necessarily translate into corresponding differences in their learned functions.

This perspective raises an interesting open question: can the co-training process be designed such that the learning trajectories of the two networks become more functionally independent? For example, it would be interesting to investigate whether increasing predictive diversity and reducing error overlap could further improve the ensemble benefit obtained from the two networks.

From this perspective, our findings do not imply that having two basins is inherently unhelpful in LNL. Rather, they suggest that the mere existence of two networks or two distinct points in parameter space does not guarantee that their ensemble will provide substantial informational diversity. In co-training, part of the computational cost of training a second network may therefore fail to translate into genuinely new and independent information. LLE suggests that, under such circumstances, targeted local exploration around a single model can recover a substantial portion of the ensemble benefit at a lower computational cost.

5 Understanding LLE in a Controlled SSL Setting

In this section, we study the behavior of LLE in a controlled semi-supervised learning setting. The objective is not to establish a new state of the art in SSL, but to understand the mechanism through which LLE improves pseudo-label generation.

5.1 Experimental Setup

We conduct the analysis on CIFAR-10 by randomly selecting 5,000 samples as labeled data and treating the remaining samples as unlabeled. A PreActResNet18 is used as the backbone network.

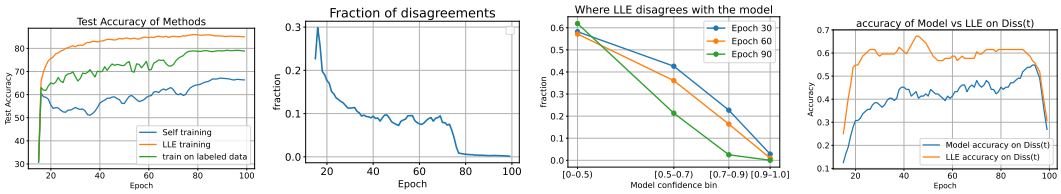
The model is first trained using only the labeled data for 15 epochs. Starting from epoch 16, unlabeled samples are incorporated through teacher-generated pseudo-labels. Training lasts for 100 epochs in total.

We use SGD with an initial learning rate of 0.02, which is reduced to 0.002 at epoch 75. The momentum and weight decay are set to 0.9 and 5×10^{-4}, respectively. Unless otherwise stated, LLE uses $K = 2$ local models, each constructed using $s = 80$ look-ahead updates on randomly sampled mini-batches from the labeled set.

5.2 Behavioral Analysis of LLE

To better understand how LLE affects pseudo-label generation, we define $Diss(t)$ as the set of unlabeled samples on which the LLE teacher and the main model produce different predictions at epoch t.

Figure 2a compares self-training and LLE-training in the controlled SSL setting. Self-training performs even worse than training solely on labeled data, a well-known consequence of confirmation bias. In contrast, LLE provides a substantial improvement, suggesting that the teacher introduces



(a) (b) (c) (d)

Figure 2: Behavioral analysis of LLE in a controlled SSL setting. (a) Test accuracy of self-training, LLE-training, and supervised training using only labeled data. (b) Fraction of unlabeled samples on which the LLE teacher disagrees with the main model. (c) Distribution of disagreement samples across confidence bins of the main model at different stages of training. (d) Prediction accuracy of the main model and the LLE teacher on the disagreement set $Diss(t)$. The results show that LLE modifies only a small subset of pseudo-labels, concentrates primarily on low-confidence samples, and provides more accurate predictions on these samples than the main model.

useful information that is not available from the model's own predictions. We next investigate the source of this improvement.

LLE focuses on hard examples. Figure 2b reports the fraction of unlabeled samples belonging to $Diss(t)$ during training. The disagreement rate remains relatively small throughout training, indicating that LLE modifies only a limited portion of the pseudo-labels generated by the main model. To characterize these disagreements, Figure 2c groups the samples in $Diss(t)$ according to the confidence of the main model. Most disagreements occur in low-confidence regions, showing that LLE primarily intervenes on samples for which the model is uncertain. These observations suggest that the additional information provided by LLE is concentrated near difficult decision boundaries rather than on easy examples whose pseudo-labels are already reliable.

LLE injects more accurate information. Generating predictions that differ from those of the main model is beneficial only if the new predictions are more accurate. To evaluate this aspect, Figure 2d compares the accuracy of the LLE teacher and the main model on the set $Diss(t)$. Across most of training, the LLE teacher achieves substantially higher accuracy on these samples than the main model. Therefore, the disagreements produced by LLE arise primarily from correcting the model's mistakes rather than introducing additional noise. This observation provides a direct explanation for the performance gains observed in Figure 2a: LLE intervenes mainly on uncertain samples and supplies more accurate pseudo-labels precisely where the model is most prone to error. This behavior differs importantly from methods such as Sohn et al. (2020); Wang et al. (2022); Zhang et al. (2021). In these methods, low-confidence samples are typically excluded from the pseudo-labeling process because the model's predictions on these points are not reliable. In contrast, rather than discarding these samples, LLE attempts to generate more accurate pseudo-labels for them and thereby injects useful information into the training process.

6 Experiments

6.1 Datasets and Implementation Details

In this section, we evaluate the effect of using LLE within the framework of state-of-the-art methods in the field of Learning with Noisy Labels. The goal of these experiments is to investigate whether LLE can replace the second network in co-training-based methods without altering the core logic of existing algorithms. To this end, we select three well-known LNL methods: DivideMix Li et al. (2020), UNICON Karim et al. (2022), and C2D Zheltonozhskii et al. (2022). All experiments are conducted on the CIFAR-10 and CIFAR-100 datasets under symmetric noise. In all cases, the network architecture, number of epochs, optimizer, and other training settings are kept exactly consistent with the original implementation of each method. In particular, in most experiments, PreActResNet18 He et al. (2016) is used as the backbone model, and the training process runs for approximately 300 epochs using SGD. The only difference from the original versions of these

Method	CIFAR-10				CIFAR-100			
	20%	50%	80%	90%	20%	50%	80%	90%
Co-teaching+Yu et al. (2019)	89.5	85.7	67.4	47.9	65.6	51.8	27.9	13.7
MixupZhang et al. (2017)	95.6	87.1	71.6	52.2	67.8	57.3	30.8	14.6
PENCILYi & Wu (2019)	92.4	89.1	77.5	58.9	69.4	57.5	31.1	15.3
Meta-LearningLi et al. (2019)	92.9	89.3	77.4	58.7	68.5	59.2	42.4	19.5
M-correctionArazo et al. (2019)	94.0	92.0	86.8	69.1	73.9	66.1	48.2	24.3
AugDescNishi et al. (2021)	96.3	95.4	93.8	91.9	79.5	77.2	66.4	41.2
GCEGhosh & Lan (2021)	90.0	89.3	73.9	36.5	68.1	53.3	22.1	8.9
Sel-CL+Li et al. (2022)	95.5	93.9	89.2	81.9	76.5	72.4	59.6	48.8
MLCZheng et al. (2021b)	92.6	88.1	77.4	67.9	66.8	52.7	21.8	15.0
MSLCWu et al. (2021)	93.4	89.9	69.8	56.1	72.5	65.4	24.3	16.7
MOIT+Ortego et al. (2021)	94.1	91.8	81.1	74.7	75.9	70.6	47.6	41.8
TCLHuang et al. (2023)	95.0	93.9	92.5	89.4	78.0	73.3	65.0	54.5
DivideMixLi et al. (2020)	**96.1**	94.6	93.2	76.0	77.3	74.6	60.2	31.5
LLE-DivideMix	96	**95.5**	**93.4**	**92.4**	**77.9**	**75.4**	**60.6**	**45.2**
UniConKarim et al. (2022)	96.0	95.6	93.9	90.8	78.9	**77.6**	63.9	44.8
LLE-UniCon	**96.5**	**96.0**	93.9	**93.3**	78.9	77.4	**66.2**	**55.4**
C2DZheltonozhskii et al. (2022)	**96.3**	95.2	94.4	93.5	**78.7**	76.4	67.8	58.7
LLE-C2D	96.1	**95.4**	94.4	**94.3**	78.6	**76.5**	**68.9**	**63.5**

Table 1: Test accuracy (%) on CIFAR-10 and CIFAR-100 under different symmetric noise ratios. Results of competing methods are taken directly from Zhou et al. (2024). For a fair comparison, we report the best test accuracy achieved during training for all methods. LLE-DivideMix, LLE-UniCon, and LLE-C2D denote the corresponding algorithms after replacing the second network with the proposed LLE teacher.

methods lies in the teacher module. In the original versions of these algorithms, two networks are trained simultaneously from the beginning, and each network uses the output of the other network for sample selection or label correction. In our proposed version, the second network is removed and replaced by an LLE-based teacher constructed at each epoch. This teacher is used at exactly the same points in the algorithm where the second network was used in the original version. In all experiments, the number of local models is set to $K = 2$. Each local model is constructed by training the main model on $s = 80$ random mini-batches sampled from the set $\mathcal{D}_{lle}^{(t)}$ at epoch t. For low and medium noise rates (20% and 50%), the set $\mathcal{D}_{lle}^{(t)}$ is selected from among the samples that the algorithm identified as clean in the previous epoch. However, at very high noise rates (80% and 90%), the quality of this set is usually insufficient for constructing local models. In this case, similar to the common practice in many LNL methods Hendrycks et al. (2018); Zheng et al. (2021a); Shu et al. (2019); Ren et al. (2018); Jiang et al. (2018); Zhou et al. (2024), we assume access to a small set of correctly labeled data. Specifically, in all experiments involving noise rates of 80% and 90%, we assume that only 2% of the entire training data is available with correct labels, and this set is used solely for constructing the local models. Note that this clean data does not enter the main training process of the algorithm and is used only for constructing the LLE-based teacher.

6.2 Experimental Results

Table 1 reports the test accuracy of different LNL methods before and after replacing their teacher module with the proposed LLE teacher. Overall, the results show that LLE consistently maintains the performance of the original methods while requiring only a single main network during training. Across most noise settings, the accuracy of the LLE-based versions remains comparable to that of their original counterparts. More importantly, under extremely noisy conditions, LLE provides a substantial performance improvement. In particular, for DivideMix, replacing the peer network with LLE yields the largest gains at very high noise rates. Under 90% symmetric noise, the LLE-based version improves the final test accuracy by approximately 14 percentage points on both CIFAR-10 and CIFAR-100. Similar trends can also be observed for UNICON and C2D, indicating that the benefit of LLE is not tied to a specific algorithm and generalizes across different LNL frameworks.

Table 2 reports the average execution time per epoch on CIFAR-10 with 20% symmetric noise, measured on an NVIDIA RTX 5090 GPU. The results demonstrate that replacing the second network with LLE dramatically reduces computational cost. The average epoch time of the LLE-based versions is approximately 63% of the original co-training implementations. This reduction is expected because the original methods require training two complete semi-supervised learning pipelines simultaneously, whereas LLE only constructs two lightweight local models using a small number of look-ahead updates from the main model. Consequently, most of the computational overhead associated with the second network is eliminated while preserving, and in some cases improving, the overall performance.

Method	Running Time(s)
DivideMix	67.8
LLE+DivideMix	**42.5**
UniCon	43
LLE+UniCon	**26.7**
C2D	39
LLE+C2D	**27.5**

Table 2: Average execution time per epoch (seconds) on CIFAR-10 with 20% symmetric noise, measured on a single NVIDIA RTX 5090 GPU. LLE substantially reduces the computational overhead of co-training by replacing the second network with lightweight local look-ahead models.

6.3 Ablation Studies

Effect of ensemble prediction, number of local models, and look-ahead steps. Table 3 presents an ablation study of LLE on top of DivideMix. Unless otherwise specified, the default configuration uses $K = 2$ local models and $s = 80$ look-ahead updates. The first two rows investigate the effect of the prediction source used at test time. The default LLE-DivideMix reports the accuracy of the ensemble composed of the main model and all local models. In contrast, "LLE Accuracy" reports only the prediction of the ensemble of local models, while "Model Accuracy" reports only the prediction of the main model. The results show that evaluating the complete ensemble generally yields the highest accuracy, indicating that both the main model and the local models contribute complementary information to the final prediction. The next rows examine the effect of the number of local models K. Using a single local model ($K = 1$) consistently produces lower performance than the default configuration. Increasing the number of local models to three ($K = 3$) leads to results that are comparable to, and in some cases slightly better than, using a single local model. However, the default setting ($K = 2$) provides the best overall performance across the evaluated datasets and noise rates. The last rows study the effect of the number of look-ahead updates s. Reducing the number of updates to $s = 40$ generally decreases performance, particularly under high noise ratios. Increasing the number of updates to $s = 120$ improves performance in several settings, suggesting that longer look-ahead trajectories can produce higher-quality local models and, consequently, a stronger LLE teacher.

Effect of the clean validation set at high noise ratios. Table 4 studies the role of the clean validation set used for constructing local models in LLE. In the default setting, a small clean validation set is used to define $\mathcal{D}_{\text{lle}}$ when the noise ratio is extremely high. In the variant denoted by "w/o validation set", $\mathcal{D}_{\text{lle}}$ is instead chosen as the set of samples identified as clean by the sample-selection module in the previous epoch. The results show a substantial performance degradation when the validation set is removed. Under 80% and 90% noise ratios, the resulting accuracy drops not only relative to LLE-DivideMix, but also below the original DivideMix baseline. This behavior can be attributed to the quality of $\mathcal{D}_{\text{lle}}$. At very high noise rates, the set of samples estimated as clean by the previous epoch often contains a considerable number of mislabeled examples. Consequently, the local models constructed from this set are less reliable, which reduces the quality of the resulting LLE teacher and negatively affects both sample selection and pseudo-label generation.

Method	CIFAR-10		CIFAR-100	
	50%	90%	50%	90%
LLE-DivideMix	95.5	92.4	75.4	45.2
LLE Accuracy	95.4	92.4	75.4	45.2
model Accuracy	94.9	**92.6**	73.1	43.6
k = 1	95.1	92.1	75.1	46.0
k = 3	95.3	92.2	75.3	**46.2**
s = 40	95.3	91.4	74.5	42.7
s = 120	**95.6**	91	**76**	45.9

Table 3: Ablation study of LLE on DivideMix. We examine the effect of the prediction source used for evaluation, the number of local models (K), and the number of look-ahead updates (s). Results are reported on CIFAR-10 and CIFAR-100 under different noise ratios.

Method	CIFAR-10		CIFAR-100	
	80%	90%	80%	90%
LLE-DivideMix	**95.5**	**92.4**	**75.4**	**45.2**
DivideMix	93.2	76	60.2	31.5
w/o validation set	92.3	70.9	57.2	27.5

Table 4: Effect of the clean validation set used for constructing local models under high noise ratios. Results are reported for DivideMix and its LLE-based variant.

7 Conclusion and Future Works

In this paper, we revisited a fundamental design choice in modern co-training-based learning with noisy labels methods: the use of a second network during training. We introduced Local Look-Ahead Ensembling (LLE), a teacher construction strategy that replaces the peer network with an ensemble of locally explored models generated from a single training trajectory. The proposed approach preserves the overall training logic of existing methods while eliminating the need to train a second network from scratch.

Experimental results on DivideMix, UniCon, and C2D demonstrate that LLE maintains performance at low and moderate noise levels and consistently improves accuracy under severe label noise. At the same time, the computational cost is significantly reduced, since constructing a small number of local models is considerably cheaper than maintaining a complete second semi-supervised learning pipeline.

Beyond its practical benefits, LLE suggests that useful supervisory signals can be extracted from the local neighborhood of a single model. Our results indicate that a substantial portion of the information traditionally provided by a second network in co-training-based LNL methods can be recovered through local exploration within a single basin of the loss landscape.

An interesting direction for future work is to investigate the use of LLE in standard semi-supervised learning settings. Although the present work focuses exclusively on learning with noisy labels, the mechanism underlying LLE is closely related to pseudo-label generation and teacher construction, making it potentially applicable to a broader class of SSL algorithms.

Another promising direction concerns the exploration of information within a single basin. LLE employs a simple strategy based on short look-ahead optimization trajectories to construct local teachers. A natural question is whether more effective exploration mechanisms can extract richer and more diverse predictive information from the same basin and consequently lead to stronger teachers. Answering this question may provide further insight into the relationship between local model diversity, pseudo-label quality, and the role of multiple networks in modern learning algorithms.

USE OF GENERATIVE AI

Generative AI tools were used solely for language editing and improving the clarity of the manuscript. All research ideas, methodological developments, experimental design, implementation, analysis, and conclusions were produced by the authors.

REFERENCES

Eric Arazo, Diego Ortego, Paul Albert, Noel O'Connor, and Kevin McGuinness. Unsupervised label noise modeling and loss correction. In *International conference on machine learning*, pp. 312–321. PMLR, 2019.

Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International joint conference on neural networks (IJCNN)*, pp. 1–8. IEEE, 2020.

Yingbin Bai, Erkun Yang, Bo Han, Yanhua Yang, Jiatong Li, Yinian Mao, Gang Niu, and Tongliang Liu. Understanding and improving early stopping for learning with noisy labels. *Advances in Neural Information Processing Systems*, 34:24392–24403, 2021.

Aritra Ghosh and Andrew Lan. Contrastive learning improves model robustness under label noise. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2697–2702. IEEE, 2021.

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

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pp. 630–645. Springer, 2016.

Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. *Advances in neural information processing systems*, 31, 2018.

Zhizhong Huang, Junping Zhang, and Hongming Shan. Twin contrastive learning with noisy labels. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11661–11670. IEEE, 2023.

Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International conference on machine learning*, pp. 2304–2313. PMLR, 2018.

Nazmul Karim, Mamshad Nayeem Rizve, Nazanin Rahnavard, Ajmal Mian, and Mubarak Shah. Unicon: Combating label noise through uniform selection and contrastive learning. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9666–9676. IEEE, 2022.

Zhanghan Ke, Daoye Wang, Qiong Yan, Jimmy Ren, and Rynson WH Lau. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6728–6736, 2019.

Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. Learning to learn from noisy labeled data. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5046–5054. IEEE, 2019.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394*, 2020.

Shikun Li, Xiaobo Xia, Shiming Ge, and Tongliang Liu. Selective-supervised contrastive learning with noisy labels. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 316–325. IEEE, 2022.

Kai Lion, Lorenzo Noci, Thomas Hofmann, and Gregor Bachmann. How good is a single basin? In *International Conference on Artificial Intelligence and Statistics*, pp. 4015–4023. PMLR, 2024.

Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *Advances in neural information processing systems*, 33:20331–20342, 2020.

Yangdi Lu and Wenbo He. Selc: self-ensemble label correction improves learning with noisy labels. *arXiv preprint arXiv:2205.01156*, 2022.

Kento Nishi, Yi Ding, Alex Rich, and Tobias Hollerer. Augmentation strategies for learning with noisy labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8022–8031, 2021.

Diego Ortego, Eric Arazo, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Multi-objective interpolation training for robustness to label noise. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6606–6615, 2021.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1944–1952, 2017.

Siyuan Qiao, Wei Shen, Zhishuai Zhang, Bo Wang, and Alan Yuille. Deep co-training for semi-supervised image recognition. In *European conference on computer vision*, pp. 142–159. Springer, 2018.

Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *International conference on machine learning*, pp. 4334–4343. PMLR, 2018.

Jun Shu, Qi Xie, Lixuan Yi, Qian Zhao, Sanping Zhou, Zongben Xu, and Deyu Meng. Meta-weight-net: Learning an explicit mapping for sample weighting. *Advances in neural information processing systems*, 32, 2019.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *Advances in neural information processing systems*, 33: 596–608, 2020.

H Song, M Kim, D Park, and JG Lee. How does early stopping help generalization against label noise? arxiv 2019. *arXiv preprint arXiv:1911.08059*, 2019.

Xinshao Wang, Yang Hua, Elyor Kodirov, David A Clifton, and Neil M Robertson. Proselflc: Progressive self label correction for training robust deep neural networks. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 752–761. IEEE, 2021.

Yidong Wang, Hao Chen, Qiang Heng, Wenxin Hou, Yue Fan, Zhen Wu, Jindong Wang, Marios Savvides, Takahiro Shinozaki, Bhiksha Raj, et al. Freematch: Self-adaptive thresholding for semi-supervised learning. *arXiv preprint arXiv:2205.07246*, 2022.

Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *2020 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 13723–13732. IEEE, 2020.

Yichen Wu, Jun Shu, Qi Xie, Qian Zhao, and Deyu Meng. Learning to purify noisy labels via meta soft label corrector. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 10388–10396, 2021.

Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. Are anchor points really indispensable in label-noise learning? *Advances in neural information processing systems*, 32, 2019.

Kun Yi and Jianxin Wu. Probabilistic end-to-end noise correction for learning with noisy labels. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7010–7018. IEEE, 2019.

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *International conference on machine learning*, pp. 7164–7173. PMLR, 2019.

Suqin Yuan, Lei Feng, and Tongliang Liu. Early stopping against label noise without validation data. In *International Conference on Learning Representations*, volume 2024, pp. 21977–21997, 2024.

Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in neural information processing systems*, 34:18408–18419, 2021.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

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

Evgenii Zheltonozhskii, Chaim Baskin, Avi Mendelson, Alex M Bronstein, and Or Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 387–397. IEEE, 2022.

Guoqing Zheng, Ahmed Hassan Awadallah, and Susan Dumais. Meta label correction for noisy label learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11053–11061, 2021a.

Guoqing Zheng, Ahmed Hassan Awadallah, and Susan Dumais. Meta label correction for noisy label learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 11053–11061, 2021b.

Yuyin Zhou, Xianhang Li, Fengze Liu, Qingyue Wei, Xuxi Chen, Lequan Yu, Cihang Xie, Matthew P Lungren, and Lei Xing. L2b: Learning to bootstrap robust models for combating label noise. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23523–23533. IEEE, 2024.

A Appendix

You may include other additional sections here.