AN INFORMATION FUSION APPROACH TO LEARNING WITH INSTANCE-DEPENDENT LABEL NOISE

Anonymous authors
Paper under double-blind review

ABSTRACT

Instance-dependent label noise (IDN) widely exists in real-world datasets and usually misleads the training of deep neural networks. Noise transition matrix (NTM) (i.e., the probability that clean labels flip into noisy labels) is used to characterize the label noise and can be adopted to bridge the gap between clean and noisy underlying data distribution. However, most of instances are long-tail, i.e., the number of appearance for each instance is usually limited, which leads to the gap between underlying distribution and empirical distribution. Therefore, the genuine problem caused by IDN is *empirical*, instead of underlying, *data distribution mismatch* during training. To directly tackle empirical distribution mismatch problem, we propose *posterior transition matrix* (PTM) to posteriorly model label noise given limited observed noisy labels achieving *statistically consistent classifiers*. Note that even if the instance is corrupted by the same NTM, the intrinsic randomness incurs to different noisy labels, and thus requires different correction methods. Motivated by this observation, we propose an **I**nformation **F**usion (IF) approach to fine-tune the NTM based on estimated PTM. Specifically, we adopt the noisy labels and model predicted probability to estimate PTM and then correct the NTM in *forward propagation*. Empirical evaluations on synthetic and real-world datasets demonstrate that our method is superior to the state-of-the-art approaches, and achieves more stable training for instance-dependent label noise.

1 INTRODUCTION

Data labeling annotated from human efforts, such as crowdsourcing (Yan et al., 2014; Chen et al., 2017) and online queries (Divvala et al., 2014), may be heavily noisy in practice. To make it worse, the label noise stemmed from the human annotations is often instance-dependent. For example, the images closing to the decision boundary are usually prone to be mislabeled (Zhang et al., 2021; Zhu et al., 2021b). On the other hand, the remarkable success of deep neural networks (DNNs) on supervised learning tasks heavily relies on the expressive power and a large number of data with accurate labels. Unfortunately, deep neural networks' ability of memorizing noisy labels leads to poor generalization (Zhang et al., 2017). It is challenging to learn with practical instance-dependent label noise (IDN) due to the hidden and complicated label noise properties (Zhu et al., 2021a; Liu, 2021).

The methods of dealing with noisy labels falls in two lines, including heuristically identifying noisy samples or statistical label noise modeling. The training of deep neural network often learns the clean labels first (Arpit et al., 2017) and memorizes the noisy labels gradually, which is recognized as the memorization effect. Based on the general memorization effects, the heuristical methods are all designed via following the anomaly detection strategy: identifying the noisy sample based on different behaviors (e.g., loss values) between clean and noisy samples during training. The typical methods contain sample selection (Yu et al., 2019; Han et al., 2018b), reweight samples (Cheng et al., 2021; Jiang et al., 2018; Ren et al., 2018), label correction (Ma et al., 2018; Tanaka et al., 2018), and regularization (Han et al., 2018a). Although these algorithms empirically work well, without modeling the label noise explicitly, the reliability cannot be guaranteed.

Another line of works relies on *noise transition matrix* (NTM) to model the label noise statistically (Xia et al., 2019; 2020; Patrini et al., 2017). Namely, the NTM quantifies the probabilities that clean labels flip into noisy labels. Although the NTM-based methods possess theoretical guarantee, under

IDN, the NTM estimation for each instance is pretty challenging. To ease the estimation, some unrealistic assumptions have to be posed the on NTM, including instance-independent transition matrix (Liu & Guo, 2020; Wei & Liu, 2021), symmetric transition matrix (Menon et al., 2018), upper bounded noise rate (Cheng et al., 2020), and part-dependent label noise (Xia et al., 2020). However, under the complex IDN, the empirical noise distribution is highly different with underlying noise distribution. For example, in Figure 1, the underlying and empirical noisy distribution for long-tail instance is different since the empirical noisy label is either the same or different with clean label. Additionally, observed noisy label provide *inductive bias* toward label corruption. In other word, the genuine problem causing from IDN is the *empirical*, instead of underlying, clean and noisy distribution mismatch problem.

To mitigate empirical distribution mismatch problem, we propose the posterior transition matrix (PTM), posteriorly modeling label noise given the observed noisy labels. We adopt PTM to provably bridge the gap between clean and noisy underlying data distribution, and anchor points (i.e., data points that belong to a specific class almost surely) empirical distribution simultaneously. We also provide a easy-to compute PTM estimation under specific condition of low label noise. To extend the applicability to a wide range of label noise, motivated by Kalman filtering (Kalman, 1960), we proposed the information fusion (IF) method linearly combining the estimated noise and PTM, which achieves lower transition matrix estimation error. We empirically show that proposed method IF can achieve higher accuracy and more stable training.

The main contributions of this work are summarized below.

- We propose a new concept, named PTM, achieving consistent classifier for underlying distribution and anchor point empirical distribution mismatch simultaneously.
- We propose the simple PTM estimation method based on observed noisy label under low label noise. To extend the applicability, we propose IF method, linearly combining the estimated NTM and PTM, to achieve lower estimation error with theoretical and experimental justification.
- We experimentally demonstrate the superiority of proposed IF method achieving higher accuracy and more stable training under synthetic and real-world label noise.

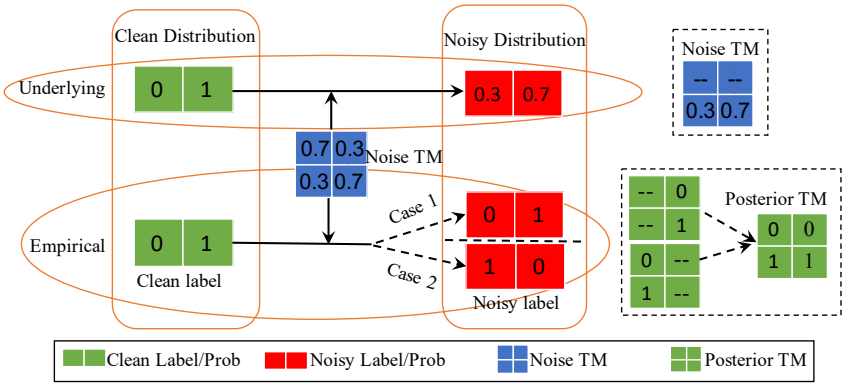


Figure 1: Illustration of the thought experiment. Assume we use *identified* noise transition matrix (PTM) to corrupt the instance with clean label $[0, 1]$. From underlying distribution perspective, the noisy underlying distribution is $[0.3, 0.7]$. From empirical distribution perspective, the label of this sample is *statistically* either the same (case 1) or different (case 2) with clean label leading to different PTM. In other words, the observed noisy labels (posterior information) provide *inductive bias* of label correction motivating the definition of PTM.

2 PRELIMINARIES

Throughout this paper, we adopt uppercase letters to denote random variables, and lowercase letters to denote particular realization of the random variables. We target the c-class classification problem with instance-dependent (label) noise (IDN). Let $\mathbb{P}(X, Y)$ be the *underlying clean distribution* of the random variables (X, Y), where X and Y represent the instance and clean label. However, in

real-world scenario, the clean label is corrupted and thus only noisy label, denoted as $\tilde{Y}$, is observed. Given a set of N training instances denoted by $\tilde{D} := \{(\boldsymbol{x}_n, \tilde{y}_n)\}_{n=1}^N$, where $\boldsymbol{x}_n$ is the instance vector of n-th sample, $\tilde{y}_n \in [c] := \{1, \cdots, c\}$ is the corresponding observed but noisy label, our goal is to predict the clean label y_n for any given instance $\boldsymbol{x}_n$.

Data distribution We assume that the unobserved clean samples $(\boldsymbol{x}_n, y_n)$ and available noisy samples $(\boldsymbol{x}_n, \tilde{y}_n)$ are drawn from unknown *underlying clean distribution* $\mathbb{P}(X, Y)$ and *underlying noisy distribution* $\mathbb{P}(X, \tilde{Y})$, respectively. For the noisy samples data $\tilde{D}$, we may approximate underlying noisy distribution by the *empirical noisy distribution*, i.e., $\hat{\mathbb{P}}_{\tilde{D}}(X, \tilde{Y}) = \frac{1}{N}\sum_{n=1}^N \delta(X = \boldsymbol{x}_n, \tilde{Y} = \tilde{y}_n)$, where $\delta(X = \boldsymbol{x}_n, \tilde{Y} = \tilde{y}_n)$ is a Dirac mass centered at $(\boldsymbol{x}_n, \tilde{y}_n)$. Since the clean data sample is unobserved, we define posterior empirical clean distribution $\hat{\mathbb{P}}(X, Y|\tilde{D})$ as the inference posterior empirical clean distribution from noisy samples data $\tilde{D}$. Based on Bayes's rule, the posterior empirical clean distribution is given by $\hat{\mathbb{P}}(X, Y|\tilde{D}) = \frac{1}{N}\sum_{n=1}^N \sum_{y_n=1}^c \delta(X = \boldsymbol{x}_n, Y = \tilde{y}_n)\mathbb{P}(Y = y_n|\tilde{Y} = \tilde{y}_n, X = \boldsymbol{x}_n)$.

Noise transition matrix $T(\boldsymbol{x})$ To describe the corruption process of the label, the NTM $T(\boldsymbol{x}) \in \mathbb{R}^{c\times c}$ is defined as $T_{i,j}(\boldsymbol{x}) = \mathbb{P}(\tilde{Y} = j|Y = i, X = \boldsymbol{x})$, which represents the transition probability of the instance $\boldsymbol{x}$ labeling from clean i-th class to noisy j-th class. The underlying noisy label $\tilde{Y}$ probability given the instance $\boldsymbol{x}$ satisfies:

$$\mathbb{P}(\tilde{Y} = j|X = \boldsymbol{x}) = \sum_{i=1}^c \mathbb{P}(\tilde{Y} = j|Y = i, X = \boldsymbol{x})\mathbb{P}(Y = i|X = \boldsymbol{x}) = \sum_{i=1}^c T_{ij}(\boldsymbol{x})\mathbb{P}(Y = i|X = \boldsymbol{x}).$$

The NTM $T(\boldsymbol{x})$ hence bridges the gap between the underlying clean labels probability $\mathbb{P}(\mathbf{Y}|X = \boldsymbol{x}) = \Big[\mathbb{P}(Y = 1|X = \boldsymbol{x}), \cdots, \mathbb{P}(Y = c|X = \boldsymbol{x})\Big]^\top$ and underlying noisy labels probability $\mathbb{P}(\tilde{\mathbf{Y}}|X = \boldsymbol{x}) = \Big[\mathbb{P}(\tilde{Y} = 1|X = \boldsymbol{x}), \cdots, \mathbb{P}(\tilde{Y} = c|X = \boldsymbol{x})\Big]^\top$ given instance $\boldsymbol{x}$, i.e., $\mathbb{P}(\tilde{\mathbf{Y}}|X = \boldsymbol{x}) = T(\boldsymbol{x})^\top \mathbb{P}(\mathbf{Y}|X = \boldsymbol{x})$.

Loss Correction method Let function $f(\cdot)$ represents a neural network and $f(\boldsymbol{x})$ denotes the c-dimensional output probability for instance $\boldsymbol{x}$, where the ith index of the output $y_{f,i}(\boldsymbol{x})$ represents the predicted probability for class i. The common approach is to minimize the cross-entropy (CE) loss $l(f(\boldsymbol{x}), y) := -\log(y_{f,y}(\boldsymbol{x}))$ to force the output $y_{f,y}(\boldsymbol{x})$ approximate 1. However, the label noise may mislead deep learning model. The existing NTM based methods first estimate NTM $T(\boldsymbol{x})$ and then adopt it to correct the loss function. For example, in the forward correction procedure (Patrini et al., 2017), the estimated NTM is adopted to corrupt the output predicted probability $f(\boldsymbol{x})$, i.e., the corrupted predicted probability is $\tilde{f}(\boldsymbol{x}) = T(\boldsymbol{x})^\top f(\boldsymbol{x})$, and then compares it with the noisy label $\tilde{y}$. Suppose $T(\boldsymbol{x})$ is non-singular and the loss function is proper and composite, the *forward loss correction* can achieve *consistent classifier*, i.e., the optimal classifier for the corrected loss with respect to the underlying noisy distribution is the same as that for the CE loss with respect to the underlying clean distribution:

$$\arg\min_f \mathbb{E}_{(X,\tilde{Y})\sim\mathbb{P}(X,\tilde{Y})}[l(\tilde{Y}, T(X)^\top f(X))] = \arg\min_f \mathbb{E}_{(X,Y)\sim\mathbb{P}(X,Y)}[l(Y, f(X))]. \tag{1}$$

3 Learning with Information Fusion

Even though the existing loss correction methods could achieve consistent classifier theoretically, their performances are still undesirable in practice. Since the deep learning model is often trained on the empirical distribution with limited samples, the NTM cannot correctly bridge the empirical clean distribution and empirical noisy distribution, i.e., $\hat{\mathbb{P}}(\tilde{Y} = j|X = \boldsymbol{x}) \neq \sum_{i=1}^c T_{ij}(\boldsymbol{x})\hat{\mathbb{P}}_{\tilde{D}}(Y = i|X = \boldsymbol{x})$. In other words, the NTM-based method may not work well in practice since due to the empirical clean and noisy distribution mismatch problem in IDN.

Aiming to utilize the observed noisy label, we propose a concept, named posterior transition matrix (PTM), to describe the transition probability given the observed noisy labels (formally defined in

Section 3.1). The motivation for PTM stems from a simple thought experiment as shown in Figure 1, which shows that PTM can model the label noise better than NTM. Assume that the neural network predicts a reasonable probability for the clean label class, it is possible to determine the PTM via the model prediction and noisy labels. Note that both the noise and PTM are unknown and we need to estimate them in a specific method, the natural question is that can we take benefit from these two estimated transition matrices? Our proposed information fusion method provides the positive answer, which can achieve better performance and more stable training.

3.1 THE LOSS CORRECTION METHOD

The main goal is to train a c-class neural network classifier $f(\boldsymbol{x}, \omega)$ to predict the clean label probability $\mathbb{P}(Y|X)$. Since only the noisy labels are observed, there is a gap between the clean and noisy label, describing via NTM (Goldberger & Ben-Reuven, 2017).

Motivated by the observed noisy labels(posterior information), we define the PTM $W(\boldsymbol{x})$ to describe the posterior clean label probability given noisy label, where

$$W_{i,j}(\boldsymbol{x}) = \mathbb{P}(Y = i|\tilde{Y} = j, X = \boldsymbol{x}), \tag{2}$$

We provide the relationship between the PTM $W(\boldsymbol{x})$ and NTM $T(\boldsymbol{x})$ via Bayes' rulei.e.,

$$W_{i,j}(\boldsymbol{x}) = \frac{\mathbb{P}(Y = i, \tilde{Y} = j|X = \boldsymbol{x})}{\mathbb{P}(\tilde{Y} = j|X = \boldsymbol{x})} = \frac{\mathbb{P}(Y = i|X = \boldsymbol{x})T_{ij}(\boldsymbol{x})}{\sum_{i=1}^{c} \mathbb{P}(Y = i|X = \boldsymbol{x})T_{ij}(\boldsymbol{x})}. \tag{3}$$

Notice that only the summation of any column is 1 for PTM $W(\boldsymbol{x})$, while the summation of any rows are 1 for NTM $T(\boldsymbol{x})$. Subsequently, we provide a loss correction method, named posterior reweight loss, via NTM.

Definition 1 (Posterior reweight loss) *Assume the model predicted probability is $f(\boldsymbol{x})$ for noisy sample $(\boldsymbol{x}, \tilde{y})$ and $W(\boldsymbol{x})$ is the NTM associated with the noisy sample. The posterior reweight loss is defined as*

$$l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big) = \sum_{i=1}^{c} W_{i,\tilde{y}}(\boldsymbol{x}) l(i, f(\boldsymbol{x})), \tag{4}$$

We next analyze the property of the posterior reweight loss and provide the theoretical justification. Specifically, we analyze the expected risk $R_{\mathbb{P}(X,\tilde{Y})}(f) = \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}[l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big)]$ and empirical risk $\hat{R}_{\hat{\mathcal{D}}}(f) = \frac{1}{N}\sum_{n=1}^{N} l_{p-rew}\Big(\tilde{y}_n, f(\boldsymbol{x}_n)\Big)$ under noisy samples dataset.

Theorem 3.1 (Statistically Consistent Classifier) *The posterior reweight loss can achieve unbiased classifier for underlying distribution and empirical distribution.*

(i) For underlying distribution, the expected risk satisfies

$$\arg\min_f \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-rew}\big(\tilde{y}, f(\boldsymbol{x})\big)\Big] = \arg\min_f \mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}(X,Y)}[l(y, f(\boldsymbol{x}))], \tag{5}$$

(ii) For the anchor points samples $(\boldsymbol{x}_{ap}, \tilde{y}_{ap})$ with underlying clean probability $\mathbb{P}(Y = y_{ap}|X = \boldsymbol{x}_{ap})$, the empirical risk satisfies $l_{p-rew}\big(\tilde{y}_{ap}, f(\boldsymbol{x}_{ap})\big) = l(y_{ap}, f(\boldsymbol{x}_{ap}))$. For the empirical distribution, the empirical risk satisfies

$$\arg\min_f \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\hat{\mathbb{P}}_{\tilde{D}}(X,\tilde{Y})}\Big[l_{p-rew}\big(\tilde{y}_n, f(\boldsymbol{x}_n)\big)\Big] = \arg\min_f \mathbb{E}_{(\boldsymbol{x},y)\sim\hat{\mathbb{P}}(X,Y|\tilde{D})}\Big[l(y_n, f(\boldsymbol{x}_n))\Big]. \tag{6}$$

We also show that the PTM can be adopted in the forward manner. The definition of posterior reweight loss is given as follows,

Definition 2 (Posterior forward loss) *Assume the model prediction probability is $f(\boldsymbol{x})$ for noisy sample $(\boldsymbol{x}, \tilde{y})$ and $W(\boldsymbol{x})$ is the NTM associated with the noisy sample. The posterior forward loss is defined as*

$$l_{p-fw}\big(\tilde{y}, f(\boldsymbol{x})\big) = l\Big(\tilde{y}, \sum_{i=1}^{c} W_{i,\tilde{y}}(\boldsymbol{x}) f_i(\boldsymbol{x})\Big), \tag{7}$$

We next analyze the property of the posterior forward loss and provide the theoretical justification.

Lemma 3.2 *The posterior forward loss is not larger than posterior reweight loss,*

$$l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big) \geq l_{p-fw}\big(\tilde{y}, f(\boldsymbol{x})\big) \tag{8}$$

Theorem 3.3 (Statistically Consistent Classifier) *Suppose the loss function $l(y, f)$ is convex with respect to f (the convex condition is commonly satisfies, e.g., cross entropy loss), and the minimum expected risk $R_{\mathbb{P}(X,Y)}(f)$ can achieve 0, then the posterior forward loss can achieve unbiased classifier for underlying distribution,*

$$\arg\min_f \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-fw}\big(\tilde{y}, f(\boldsymbol{x})\big)\Big] = \arg\min_f \mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}(X,Y)}[l(y, f(\boldsymbol{x}))]. \tag{9}$$

Compare the posterior forward loss with forward loss, it is seen that both of them correct the output of the neural network with statistically consistent risk guarantee, while the difference falls in the different transition matrix in the loss correction. In addition, the statistically consistent risk for forward loss and posterior forward loss both require the accurate transition matrix estimation. However, the label/PTM estimation is pretty challenging since the label noise is instance-dependent in real world. Such observation motivates us to combine the NTM $T(\boldsymbol{x})$ and PTM $W(\boldsymbol{x})$ to correct the loss.

Motivated by Kalman filtering(Kalman, 1960), a famous estimation method combining the prior knowledge and measurement information, we propose an information approach to tackle instance-dependent label noise via loss correction. Specifically, we firstly claim that there are prior knowledge information and measurement information on the label noise, where prior knowledge information and measurement information represent the estimated NTM and PTM.

3.2 PTM Estimation

The posterior loss correction requires knowing PTM $W(\boldsymbol{x})$ given any instance $\boldsymbol{x}$. However, the PTM is unknown and needs to be estimated. In this subsection, we provide a simple PTM estimation method under the *condition* that the neural network output $f(X)$ probability can well approximate the underlying probability $\mathbb{P}(Y|X)$, i.e., $f(\boldsymbol{x}) \approx W(X=\boldsymbol{x})\hat{\mathbb{P}}(\tilde{\mathbf{Y}}|X=\boldsymbol{x})$. Intuitively, this condition is strong since neural network is prone to overfit noisy data. To achieve this condition, the warm-up training strategy and iterative PTM estimation are adopted. Warm-up training to make "good" neural network output approximating the empirical clean distribution since the network fits clean samples in the beginning. Subsequently, we iteratively estimate the PTM for loss correction during training.

For the case that the number of appearance for instance $\boldsymbol{x}$ is large, the empirical noisy distribution convergences to the underlying noisy distribution, i.e., $\hat{\mathbb{P}}(\tilde{\mathbf{Y}}|X=\boldsymbol{x}) \longrightarrow \mathbb{P}(\tilde{\mathbf{Y}}|X=\boldsymbol{x})$. For the long-tail instance with only one appearance $(\boldsymbol{x}, \tilde{y})$, the empirical noisy labels distribution is always one-hot. Thus, this condition requires that the output of neural network $f(\boldsymbol{x})$ should approximate to the underlying posterior clean probability, which is quite strong.

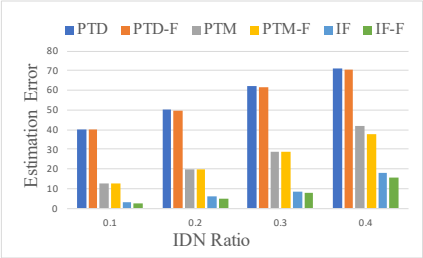


Figure 2: Estimation error for CIFAR10

Under this condition, we propose a PTM estimation method based on the noisy labels and the model predicted probability. For the long-tail instance with only one appearance $(\boldsymbol{x}, \tilde{y})$, the observed noisy labels distribution is the one-hot. Compared with the NTM estimation (Xia et al., 2019; 2020), the proposed estimation method does not require the anchor point assumption that data points belong to a specific class almost surely. Specifically, the PTM can be estimated via the following formulation

$$\hat{W}_{i,j}(\boldsymbol{x}) = \begin{cases} f_i(\boldsymbol{x}), & \text{if } \tilde{y} = j; \\ 0, & \text{if } \tilde{y} \neq j. \end{cases} \tag{10}$$

For the case that the number of appearance is larger than one, the PTM estimation is ill-posed since the number of unknown variable c^2 is larger the number of constraint c. Thus, we choose the PTM

with minimum Frobenius norm, which is consistent with that for the case of only one appearance instance. Specifically, we have the following theorem:

Theorem 3.4 (Posterior transition matrix estimation) *we aim to find the solution with minimal norm solution, i.e.,*

$$\hat{W}(\boldsymbol{x}) = \arg\min_{W} ||W||_F^2$$
$$s.t. W^\top(\boldsymbol{x})\hat{\mathbb{P}}(\tilde{Y}|\boldsymbol{x}) = f(\boldsymbol{x}), \textit{ for } \forall \boldsymbol{x}$$

Then, we have the optimal solution as follows,

$$\hat{W}(\boldsymbol{x}) = \frac{f(\boldsymbol{x})\hat{\mathbb{P}}(\tilde{Y}|\boldsymbol{x})^\top}{||\hat{\mathbb{P}}(\tilde{Y}|\boldsymbol{x})||_2^2}. \tag{11}$$

Theorem 3.4 provides the PTM estimation for general empirical noisy labels distribution. For the case of only one appearance instance $\boldsymbol{x}$, the empirical noisy distribution satisfies $||\hat{\mathbb{P}}(\tilde{Y}|\boldsymbol{x})||_2 = 1$, and estimated PTM in equation (11) is consistent with that in equation (10).

3.3 Information Fusion

Subsection 3.2 introduces the PTM estimation methods based on the observed noisy label, the condition that neural network approximates clean label is quite strong even though the warm-up strategy and iterative estimation are adopted and thus PTM estimation error could be large for large IDN rate. To further reduce the estimation error, motivated by Kalman filtering, we propose the information fusion (IF) to obtain more accurate transition matrix estimation via weighted average PTM estimation and NTM estimation with intuitive and theoretical justification. Intuitively, for each instance, the estimated NTM and PTM may have different estimation accuracy. Via adaptively and linearly combine these two matrices, it is possible to obtain a more accurate transition matrix estimation. Specifically, we quantify the estimation uncertainty and lower uncertainty incurs higher weight. In this way, more accurate estimated transition matrix can be provided. Before the illustration of information fusion algorithm, we need to quantify the transition matrix estimation error without ground-truth. Suppose the (posterior) forward loss is adopted. For the estimated NTM $\hat{T}(\boldsymbol{x})$, the corrupted model predicted probability $\tilde{f}(\boldsymbol{x}) = \hat{T}^\top(\boldsymbol{x})f(\boldsymbol{x})$. Suppose the corrupted model predicted probability is accurate, the noisy label $\tilde{Y}$ satisfies c-dimension Bernoulli distribution with parameter $\tilde{f}(\boldsymbol{x})$, i.e., $\tilde{Y} \sim Bernoulli\big(\hat{T}^\top(\boldsymbol{x})f(\boldsymbol{x})\big)$, where $Bernoulli(\cdot)$ represents multi-dimension Bernoulli distribution. Furthermore, we define the uncertainty $\sigma_{\hat{T}}$ for transition matrix $\hat{T}(\boldsymbol{x})$ as the trace of the covariance matrix for $\tilde{Y}$, i.e., $\sigma_{\hat{T}}(\boldsymbol{x}) = Tr(K_{\tilde{Y}}(\boldsymbol{x}))$, where $Tr(\cdot)$ means the trace of matrix, $K_{\tilde{Y}}(\boldsymbol{x})$ represents the covariance matrix of random variable $\tilde{Y}$. For the multi-dimension Bernoulli distribution, we have the covariance matrix as follows,

$$K_{\tilde{Y}} = \mathbb{E}[(\tilde{Y} - \mathbb{E}[\tilde{Y}])(\tilde{Y} - \mathbb{E}[\tilde{Y}])^\top] = \text{diag}(\hat{T}^\top(\boldsymbol{x})f(\boldsymbol{x})) - \hat{T}^\top(\boldsymbol{x})f(\boldsymbol{x})f(\boldsymbol{x})^\top\hat{T}(\boldsymbol{x})$$

Taking trace operation on the covariance matrix, we have the uncertainty for NTM as $\sigma_{\hat{T}}(\boldsymbol{x}) = ||\hat{T}^\top(\boldsymbol{x})f(\boldsymbol{x})||_1 - ||\hat{T}^\top(\boldsymbol{x})f(\boldsymbol{x})||_2$. Similarly, the uncertainty for the PTM estimation is given by $\sigma_{\hat{W}}(\boldsymbol{x}) = ||\hat{W}^\top(\boldsymbol{x})f(\boldsymbol{x})||_1 - ||\hat{W}^\top(\boldsymbol{x})f(\boldsymbol{x})||_2$. Once the uncertainty for estimated transition matrix has been established, we further integrate the two estimated transition matrices into Kalman transition matrix, defined as $W_{km}(\boldsymbol{x})$, via weighted average operation. Mathematically, Kalman transition matrix is given by

$$W_{km}(\boldsymbol{x}) = (1 - \lambda(\boldsymbol{x}))\hat{T}(\boldsymbol{x}) + \lambda(\boldsymbol{x})\hat{W}(\boldsymbol{x}) \tag{12}$$

where the Kalman gain $\lambda(\boldsymbol{x})$ is carefully selected to minimize the estimation error. Suppose the true PTM is $W^*(\boldsymbol{x})$, where $W^*_{ij}(\boldsymbol{x}) = 1$ if $y = i$ and $\tilde{y} = j$, otherwise 0. Subsequently, we define the reconstruction error for NTM and PTM as $e_T(\boldsymbol{x}) = W^*(\boldsymbol{x}) - \hat{T}(\boldsymbol{x})$ and $e_W(\boldsymbol{x}) = W^*(\boldsymbol{x}) - \hat{W}(\boldsymbol{x})$, respectively. The parameter $\lambda(\boldsymbol{x})$ is determined via minimization of the mean square reconstruction error. We theoretically justify the superiority of IF in a special case scenario.

Theorem 3.5 (Theoretical justification of IF) *Suppose the reconstruction error $e_T(\boldsymbol{x})$ and $e_W(\boldsymbol{x})$ is independent and the trace of covariance matrix for error $e_{\hat{T}}(\boldsymbol{x})$ and $e_{\hat{W}}$ can be quantified by*

$\sigma_{\hat{T}}(\boldsymbol{x})$ and $\sigma_{\hat{W}}(\boldsymbol{x})$, i.e., $Tr(e_{\hat{T}}(\boldsymbol{x})e_{\hat{T}}^{\top}(\boldsymbol{x})) = \sigma_{\hat{T}}(\boldsymbol{x})$ and $Tr(e_{\hat{W}}(\boldsymbol{x})e_{\hat{W}}^{\top}(\boldsymbol{x})) = \sigma_{\hat{W}}(\boldsymbol{x})$. For the Kalman transition matrix W_{km}, we define the reconstruction error as $\mathcal{L}_{mse}(\lambda(\boldsymbol{x})) = Tr\Big((W^*(\boldsymbol{x}) - W_{km}(\boldsymbol{x}))(W^*(\boldsymbol{x}) - W_{km}(\boldsymbol{x}))^{\top}\Big)$, then the optimal kalman gain to minimize the reconstruction error is given by

$$\lambda^*(\boldsymbol{x}) \quad = \quad \arg\min_{\lambda} \mathcal{L}_{mse}(\lambda) = \frac{\sigma_{\hat{T}}(\boldsymbol{x})}{\sigma_{\hat{T}}(\boldsymbol{x}) + \sigma_{\hat{W}}(\boldsymbol{x})} \tag{13}$$

and the minimum reconstruction error $\mathcal{L}_{mse}(\lambda^*(\boldsymbol{x})) = \frac{\sigma_{\hat{T}}(\boldsymbol{x})\sigma_{\hat{W}}(\boldsymbol{x})}{\sigma_{\hat{T}}(\boldsymbol{x})+\sigma_{\hat{W}}(\boldsymbol{x})} \leq \min(\sigma_{\hat{T}}(\boldsymbol{x}), \sigma_{\hat{W}}(\boldsymbol{x}))$

Theorem 3.5 implies that the Kalman transition matrix $W_{km}(\boldsymbol{x})$, lienarly conbining PTM and NTM, provably achieve lower estimation error under a special condition.

4 EXPERIMENTS

In this section, we empirically evaluate the effectiveness and stability of information fusion on synthetic and real-world noisy datasets. We aim to answer two questions as follows. **Q1**: Compared with the state-of-the-art transition matrix based methods, can information fusion achieve higher accuracy? **Q2**: Can information fusion leads to more accurate and robust transition matrix estimation and more stable training?

Table 1: Means and standard deviations (percentage) of classification accuracy with different instance-dependent label noise levels. Methods with "-F" adopts the Forward correction loss; methods with "-V" means that the transition matrices are revised via slack variable trick.

Dataset	Method	IDN-10%	IDN-20%	IDN-30%	IDN-40%	IDN-50%.
SVHN	CE	90.77 ± 0.45	90.23 ± 0.62	86.33 ± 1.34	65.66 ± 1.65	48.01 ± 4.59
	DMI	93.51 ± 1.09	93.22 ± 0.62	91.78 ± 1.54	69.34 ± 2.45	48.93 ± 2.34
	Forward	90.89 ± 0.60	90.65 ± 0.27	87.32 ± 0.59	78.46 ± 2.58	46.27 ± 3.90
	Reweight	92.49 ± 0.44	91.09 ± 0.34	90.25 ± 0.77	84.48 ± 0.86	45.46 ± 3.56
	T-Revision	94.24 ± 0.53	94.00 ± 0.88	93.01 ± 0.83	88.63 ± 1.37	49.02 ± 4.33
	PTD-F	91.92 ± 0.87	90.21 ± 1.01	87.11 ± 1.43	81.10 ± 2.78	67.81 ± 5.41
	PTD-F-V	92.64 ± 0.81	93.94 ± 0.25	92.71 ± 0.44	90.71 ± 0.91	80.35 ± 5.46
	PTM-F	93.78 ± 0.12	92.55 ± 0.40	89.25 ± 0.09	81.89 ± 2.82	67.30 ± 3.07
	PTM-F-V	93.89 ± 0.10	92.72 ± 0.18	88.97 ± 0.26	81.14 ± 2.06	68.95 ± 3.02
	IF-F	$\mathbf{94.70 \pm 0.14}$	$\mathbf{94.01 \pm 0.31}$	$\mathbf{93.33 \pm 0.15}$	91.60 ± 0.29	85.57 ± 3.17
	IF-F-V	94.63 ± 0.12	93.92 ± 0.26	93.21 ± 0.20	$\mathbf{91.66 \pm 0.23}$	$\mathbf{86.24 \pm 2.61}$
CIFAR10	CE	74.49 ± 0.29	68.21 ± 0.72	60.48 ± 0.62	49.84 ± 1.27	38.86 ± 2.71
	DMI	75.02 ± 0.45	69.89 ± 0.33	61.88 ± 0.64	51.23 ± 1.18	41.45 ± 1.97
	Forward	73.45 ± 0.23	68.99 ± 0.62	60.21 ± 0.75	47.17 ± 2.96	40.75 ± 2.09
	Reweight	74.55 ± 0.23	68.42 ± 0.75	62.58 ± 0.46	50.12 ± 0.96	41.08 ± 2.45
	T-Revision	74.61 ± 0.39	69.32 ± 0.64	64.09 ± 0.37	50.38 ± 0.87	42.57 ± 3.27
	PTD-F	79.77 ± 0.91	74.96 ± 0.71	70.68 ± 0.81	61.92 ± 1.59	45.34 ± 4.67
	PTD-F-V	80.08 ± 0.86	74.67 ± 0.36	71.66 ± 1.05	62.45 ± 1.73	46.16 ± 4.48
	PTM-F	78.16 ± 0.36	74.81 ± 0.81	70.03 ± 0.38	63.48 ± 0.38	51.03 ± 3.08
	PTM-F-V	78.58 ± 0.31	$75.06 \pm 0.56 \pm$	69.83 ± 0.58	62.69 ± 0.69	50.53 ± 3.41
	IF-F	80.92 ± 0.28	$\mathbf{79.58 \pm 0.52}$	74.34 ± 0.86	$\mathbf{68.21 \pm 2.21}$	50.07 ± 3.95
	IF-F-V	$\mathbf{80.94 \pm 0.43}$	79.54 ± 0.45	$\mathbf{74.67 \pm 0.92}$	68.03 ± 2.90	$\mathbf{52.34 \pm 1.31}$
F-MNIST	CE	88.54 ± 0.31	88.38 ± 0.42	84.22 ± 0.35	68.86 ± 0.78	51.42 ± 0.66
	DMI	91.98 ± 0.62	90.33 ± 0.21	84.81 ± 0.44	69.01 ± 1.87	51.64 ± 1.78
	Forward	89.05 ± 0.43	88.61 ± 0.43	84.27 ± 0.46	70.25 ± 1.28	57.33 ± 3.75
	Reweight	90.33 ± 0.27	89.70 ± 0.35	87.04 ± 0.35	80.29 ± 0.89	65.27 ± 1.33
	T-Revision	91.56 ± 0.31	90.68 ± 0.66	89.46 ± 0.45	87.21 ± 1.20	74.22 ± 0.81
	PTD-F	91.13 ± 0.34	89.51 ± 0.65	89.01 ± 0.40	89.18 ± 0.20	75.37 ± 3.63
	PTD-F-V	91.71 ± 0.28	91.18 ± 0.30	90.62 ± 0.08	89.38 ± 0.69	77.80 ± 7.29
	PTM-F	91.39 ± 0.45	90.66 ± 0.07	88.48 ± 0.09	83.20 ± 1.02	63.85 ± 3.92
	PTM-F-V	91.73 ± 0.32	91.07 ± 0.14	88.87 ± 0.62	82.20 ± 0.92	64.23 ± 4.73
	IF-F	91.58 ± 0.16	91.23 ± 0.09	90.10 ± 0.74	89.37 ± 0.26	82.78 ± 0.21
	IF-F-V	$\mathbf{92.08 \pm 0.06}$	$\mathbf{91.73 \pm 0.19}$	$\mathbf{90.82 \pm 0.33}$	$\mathbf{90.32 \pm 0.12}$	$\mathbf{83.33 \pm 0.73}$

4.1 EXPERIMENTAL SETTINGS

Datasets We verify the superiority of our approach on the three manually corrupted datasets, i.e., F-MNIST, SVHN, CIFAR-10, and one real-world noisy dataset Clothing1M. The first three datasets contain clean data and we manually corrupted the label of training dataset following from (Xia et al., 2020). More details about this instance-dependent label noise generation approach can be found in Appendix. IDN-τ means that the controlled noise rate is τ. All experiments on those datasets with synthetic instance-dependent label noise are repeated five times. The real-world dataset Clothing1M has $1M$ images with real-world noisy labels and $10k$ images with clean labels for testing. In experiments, we leave out 10% of the noisy training samples as a noisy validation set for model selection.

Baselines We compare the proposed IF method with (i) CE, which trains the standard deep network with the cross entropy loss on noisy datasets. (ii) DMI, a novel information-theoretic robust loss function for instance-independent label noise (Xu et al., 2019). (iii) Forward (Patrini et al., 2017), Reweight (Liu & Tao, 2015), and T-Revision (Xia et al., 2019). (iv) Part-dependent transition matrix (PTD) (Xia et al., 2020). These approaches utilize the NTM to correct the loss function.

Implementations Following the previous settings, we choose the neural network and optimizer as follows. We use a ResNet-18 network for F-MNIST, a ResNet-34 network for SVHN and CIFAR-10. For the optimization, we first use SGD with 0.9 momentum, 10^{-4} weight decay, 128 batch size, 50 epochs and an initial learning rate of 10^{-2} to initialize the network. Then, we adopt Adam optimizer and 5×10^{-7} learning rate to learn the NTM following PTD (Xia et al., 2020). Once the NTM is obtained, we retrain the neural network. In the forward propagation, Kalman transition matrix is obtained and then adopted to correction the loss function. In addition, the revision trick (Xia et al., 2019) is also adopted for our proposed IF method, which introduces the slack variables ΔW for the estimated Kalman transition matrix. Specifically, the slack variable ΔW is initialized as all zero elements in the experiments and can be optimized during the training. Aiming to guarantee valid transition matrix, We first project their negative entries of $W_{km}(\boldsymbol{x}) + \Delta W$ to zero and then normalize the summation of each row as 1. Following (Xia et al., 2020), we do not use any data augmentation technique in the experiments. For Clothing1M, we use a ResNet-50 pre-trained model on ImageNet. We do not use any clean data to learn the transition matrices and classifiers. After the NTM $\hat{T}(\boldsymbol{x})$ is obtained according to (Xia et al., 2020), we use SGD with 0.9 momentum, 10^{-3} weight decay, 32 batch size, and run with 10^{-3} learning rate and 10 epochs. For the evaluation, we adopt the test accuracy of the epoch with best accuracy on validation dataset.

4.2 EXPERIMENTAL RESULTS

Accuracy on synthetic and real-world noisy datasets To answer **Q1**, Table 1 reports the classification accuracy on the *SVHN*, *CIFAR-10* and *F-MNIST* with noise rate IDN-10% to IDN-50%, where the highest accuracy is bold faced. It is seen that information fusion achieves better performance compared with all baselines across the three datasets and five noise rate. For example, in *CIFAR-10*, the improvements over the strongest baseline is 1.07%, 0.83%, 4.03%, 9.22%, 9.89% for noise rate IDN-10% to IDN-50%, respectively. The higher accuracy implies that the posterior information is more important in higher noise rate. In addition, the methods IF-F and IF-F-V is comparable across the three datasets and noise rate, which demonstrates that the revision on the transition matrix is not significant beneficial compared with the method PTD. This observation validates the effectiveness of posterior information on transition matrix modification. For the real-world dataset, IF outperforms the baselines as shown in Table 2.

Transition matrix estimation evaluation during We study transition matrix estimation error in the training datat for IF and PTM to answer **Q2**. Figures 3 shows the training accuracy and estimation error of each epoch on CIFAR10 dataset with noise rate IDN-10% to IDN-40%, respectively. The shadow area represents the standard deviation. It is seen that, on all IDN rate, IF can achieve higher training accuracy and lower estimation error compared with IF.

Stability on synthetic noisy datasets We study the training stability via inspecting the test accuracy over epochs to answer **Q2**. Figures 4 shows the test accuracy with respect to epochs on CIFAR10

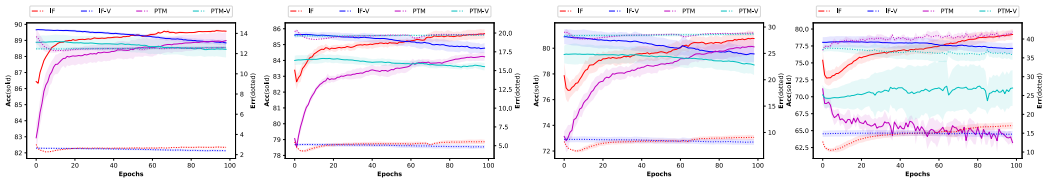


Figure 3: The training accuracy and transition matrix estimation error on CIFAR-10 datasets with IDN nose 0.1, 0.2, 0.3, and 0.4.

Table 2: Classification accuracy on *Clothing1M*. In the experiments, only noisy samples are exploited to train and validate the deep model.

CE	Forward	Reweight	T-Revision	PTD-F	PTD-F-V	IF-F	IF-F-V
68.94	69.98	70.82	71.27	70.37	70.62	72.04	**72.29**

dataset with noise rate IDN-10% to IDN-40%, respectively. The shadow area represents the standard deviation of the accuracy on the same epoch across five experiments. We can clearly observe that, on both low-level and high level noise, IF shows higher accuracy in a more stable way since the posterior information of each instance provides reliable guideness on the loss correction.

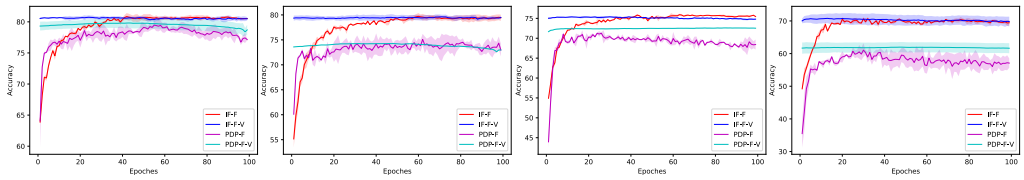


Figure 4: The test accuracy on CIFAR-10 datasets with IDN nose 0.1, 0.2, 0.3, and 0.4.

5 Conclusions

This paper presents PTM to characterize label noise given observed noisy label, and a simple but effective learning paradigm named information fusion, which trains deep neural networks under instance-dependent label noise. The main goal of PTM is to characterize the label noise via limited observed noisy labels for each instance, which can bridge the empirical noisy and posterior empirical clean distribution. Aiming to effectively combine NTM and PTM, we propose PTM to characterize label noise given observed noisy label. Furthermore, we proposed an information method integrating both transition matrix to correct the loss function. Experiments on synthetic noisy and real-world dataset show that IF can achieve higher accuracy and more stable training compared with state-of-the-art methods, especially for high noise rate. In the future, we can extend to incorporate more prior knowledge of the transition matrix, e.g., sparsity or low rank, into the end-to-end learning algorithm.

References

Anelia Angelova, Yaser Abu-Mostafam, and Pietro Perona. Pruning training sets for learning of object categories. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pp. 494–501. IEEE, 2005.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, pp. 233–242, 2017.

Antonin Berthon, Bo Han, Gang Niu, Tongliang Liu, and Masashi Sugiyama. Confidence scores make instance-dependent label-noise learning possible. In *International Conference on Machine Learning*. PMLR, 2021.

Carla E Brodley and Mark A Friedl. Identifying mislabeled training data. *Journal of artificial intelligence research*, 11:131–167, 1999.

Guangyong Chen, Shengyu Zhang, Di Lin, Hui Huang, and Pheng Ann Heng. Learning to aggregate ordinal labels by maximizing separating width. In *International Conference on Machine Learning*, pp. 787–796. PMLR, 2017.

Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. Learning with instance-dependent label noise: A sample sieve approach. In *In Proceedings of the 9th International Conference on Learning Representation*, 2021.

Jiacheng Cheng, Tongliang Liu, Kotagiri Ramamohanarao, and Dacheng Tao. Learning with bounded instance and label-dependent label noise. In *International Conference on Machine Learning*, pp. 1789–1799. PMLR, 2020.

Santosh K Divvala, Ali Farhadi, and Carlos Guestrin. Learning everything about anything: Webly-supervised visual concept learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3270–3277, 2014.

Jacob Goldberger and Ehud Ben-Reuven. Training deep neural-networks using a noise adaptation layer. 2017.

Bo Han, Jiangchao Yao, Gang Niu, Mingyuan Zhou, Ivor W Tsang, Ya Zhang, and Masashi Sugiyama. Masking: a new perspective of noisy supervision. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 5841–5851, 2018a.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W Tsang, and Masashi Sugiyama. Co-teaching: robust training of deep neural networks with extremely noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 8536–8546, 2018b.

E. Hosseini-Asl, J. M. Zurada, and O. Nasraoui. Deep learning of part-based representation of data using sparse autoencoders with nonnegativity constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 27(12):2486–2498, 2016.

Jinchi Huang, Lie Qu, Rongfei Jia, and Binqiang Zhao. O2u-net: A simple noisy label detection approach for deep neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3326–3334, 2019.

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.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

Sofia Ira Ktena, Alykhan Tejani, Lucas Theis, Pranay Kumar Myana, Deepak Dilipkumar, Ferenc Huszár, Steven Yoo, and Wenzhe Shi. Addressing delayed feedback for continuous training with neural networks in ctr prediction. In *Proceedings of the 13th ACM Conference on Recommender Systems*, pp. 187–195, 2019.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations*, 2019.

Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2015.

Yang Liu. The importance of understanding instance-level noisy labels. In *International Conference on Machine Learning*. PMLR, 2021.

Yang Liu and Hongyi Guo. Peer loss functions: Learning from noisy labels without knowing noise rates. In *International Conference on Machine Learning*, pp. 6226–6236. PMLR, 2020.

Xingjun Ma, Yisen Wang, Michael E Houle, Shuo Zhou, Sarah Erfani, Shutao Xia, Sudanthi Wijewickrema, and James Bailey. Dimensionality-driven learning with noisy labels. In *International Conference on Machine Learning*, pp. 3355–3364. PMLR, 2018.

Naresh Manwani and PS Sastry. Noise tolerance under risk minimization. *IEEE transactions on cybernetics*, 43(3):1146–1151, 2013.

Aditya Krishna Menon, Brendan Van Rooyen, and Nagarajan Natarajan. Learning from binary labels with instance-dependent noise. *Machine Learning*, 107(8):1561–1595, 2018.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep Ravikumar, and Ambuj Tewari. Learning with noisy labels. In *NIPS*, volume 26, pp. 1196–1204, 2013.

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 *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2233–2241. IEEE, 2017.

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.

Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. Joint optimization framework for learning with noisy labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5552–5560, 2018.

Dapeng Tao, Dacheng Tao, Xuelong Li, and Xinbo Gao. Large sparse cone non-negative matrix factorization for image annotation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(3):1–21, 2017.

Jiaheng Wei and Yang Liu. When optimizing f-divergence is robust with label noise. 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*, 2019.

Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. *Advances in Neural Information Processing Systems*, 33, 2020.

Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise. In *NeurIPS*, pp. 6222–6233, 2019.

Yan Yan, Rómer Rosales, Glenn Fung, Ramanathan Subramanian, and Jennifer Dy. Learning from multiple annotators with varying expertise. *Machine learning*, 95(3):291–327, 2014.

Yu Yao, Tongliang Liu, Bo Han, Mingming Gong, Gang Niu, Masashi Sugiyama, and Dacheng Tao. Towards mixture proportion estimation without irreducibility. *arXiv preprint arXiv:2002.03673*, 2020.

Shota Yasui, Gota Morishita, Fujita Komei, and Masashi Shibata. A feedback shift correction in predicting conversion rates under delayed feedback. In *Proceedings of The Web Conference 2020*, pp. 2740–2746, 2020.

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.

Xiyu Yu, Tongliang Liu, Mingming Gong, Kayhan Batmanghelich, and Dacheng Tao. An efficient and provable approach for mixture proportion estimation using linear independence assumption. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4480–4489, 2018.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. 2017.

Yikai Zhang, Songzhu Zheng, Pengxiang Wu, Mayank Goswami, and Chao Chen. Learning with feature-dependent label noise: A progressive approach. In *In Proceedings of the 9th International Conference on Learning Representation*, 2021.

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

Zhaowei Zhu, Tongliang Liu, and Yang Liu. A second-order approach to learning with instance-dependent label noise. 2021a.

Zhaowei Zhu, Yiwen Song, and Yang Liu. Clusterability as an alternative to anchor points when learning with noisy labels. In *International Conference on Machine Learning*. PMLR, 2021b.

In this Appendix, we provide A) theoretical analysis, B) Reproducibility, C) Additional Experimental Results, D) Related Works.

A THEORETICAL ANALYSIS

A.1 PROOF OF THEOREM 3.1

Based on the definition of posterior reweight loss, we have $l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big) = \sum_{y=1}^{c} \mathbb{P}(Y = y|\tilde{Y} = \tilde{y}, X = \boldsymbol{x}) l(y, f(\boldsymbol{x}))$. For noisy underlying distribution, the expected risk can be calculated as follows,

$$\begin{aligned}
& \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big)\Big] \\
= & \int_{\boldsymbol{x}} \sum_{\tilde{y}=1}^{c} \mathbb{P}(X = \boldsymbol{x}, \tilde{Y} = \tilde{y}) l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big) \mathrm{d}\boldsymbol{x} \\
= & \int_{\boldsymbol{x}} \sum_{\tilde{y}=1}^{c} \mathbb{P}(X = \boldsymbol{x}, \tilde{Y} = \tilde{y}) \sum_{y=1}^{c} \mathbb{P}(Y = y|\tilde{Y} = \tilde{y}, X = \boldsymbol{x}) l\Big(y, f(\boldsymbol{x})\Big) \mathrm{d}\boldsymbol{x} \\
= & \int_{\boldsymbol{x}} \sum_{\tilde{y}=1}^{c} \sum_{y=1}^{c} \mathbb{P}(X = \boldsymbol{x}, Y = y, \tilde{Y} = \tilde{y}) l\Big(y, f(\boldsymbol{x})\Big) \mathrm{d}\boldsymbol{x} \\
= & \int_{\boldsymbol{x}} \sum_{y=1}^{c} \mathbb{P}(X = \boldsymbol{x}, Y = y) l\Big(y, f(\boldsymbol{x})\Big) \mathrm{d}\boldsymbol{x} = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}(X,Y)}\Big[l\Big(y, f(\boldsymbol{x})\Big)\Big], \qquad (14)
\end{aligned}$$

which implies that posterior reweight loss over noisy underlying distribution equals to cross entropy loss over clean underlying distribution and thus posterior reweight loss induces to statistically consistent classifier.

For the anchor points samples $(\boldsymbol{x}_{ap}, \tilde{y}_{ap})$ with underlying clean probability $\mathbb{P}(Y = y_{ap}|X = \boldsymbol{x}_{ap})$, based on equation (3), the PTM value satisfies $W_{y_{ap},\tilde{y}_{ap}}(\boldsymbol{x}) = 1$ and $W_{k,\tilde{y}_{ap}} = 0$ for any $k \neq y_{ap}$, which implies that empirical distribution mismatch on anchor points samples can be completely mitigated. As for empirical distribution, the empirical risk satisfies

$$l_{p-rew}\big(\tilde{y}_{ap}, f(\boldsymbol{x}_{ap})\big) = l(y_{ap}, f(\boldsymbol{x}_{ap})). \qquad (15)$$

For noisy empirical distribution, the expected risk satisfies

$$\begin{aligned}
\mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\hat{\mathbb{P}}_{\tilde{D}}(X,\tilde{Y})}\Big[l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big)\Big] &= \frac{1}{N}\sum_{n=1}^{N} l_{p-rew}\Big(\tilde{y}_i, f(\boldsymbol{x}_i)\Big) \\
&= \frac{1}{N}\sum_{n=1}^{N}\sum_{y=1}^{c} \mathbb{P}(Y = y|\tilde{Y} = \tilde{y}_n, X = \boldsymbol{x}_n) l(y, f(\boldsymbol{x})) \\
&= \mathbb{E}_{(\boldsymbol{x},y)\sim\hat{\mathbb{P}}(X,Y|\tilde{D})}\Big[l(y_n, f(\boldsymbol{x}_n))\Big]. \qquad (16)
\end{aligned}$$

which implies that posterior reweight loss over noisy empiricial distribution equals to cross entropy loss over posterior clean empirical distribution.

A.2 PROOF OF LEMMA 3.2

Note that the loss function $l(y, f)$ is convex with respect to f and $\sum_{i=1}^{c} W_{i,\tilde{y}}(\boldsymbol{x}) = \sum_{i=1}^{c} \mathbb{P}(Y = i|\tilde{Y} = \tilde{y}, X = \boldsymbol{x}) = 1$ for any $\tilde{y}$ and $\boldsymbol{x}$, according to Jensen's inequality, we have

$$\begin{aligned}
l_{p-rew}\Big(\tilde{y}, f(\boldsymbol{x})\Big) &= \sum_{y=1}^{c} W_{i,\tilde{y}}(\boldsymbol{x}) l(y, f(\boldsymbol{x})) \\
&\geq l\Big(\tilde{y}, \sum_{i=1}^{c} W_{i,\tilde{y}}(\boldsymbol{x}) f(\boldsymbol{x})\Big) = l_{p-fw}\big(\tilde{y}, f(\boldsymbol{x})\big), \qquad (17)
\end{aligned}$$

which shows the relation between posterior reweight loss and posterior forward loss.

A.3 PROOF OF THEOREM 3.3

Based on Lemma 3.2 and Theorem 3.1, we have the inequality on the expected risk as follows,

$$\mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-fw}\big(\tilde{y},f(\boldsymbol{x})\big)\Big] \leq \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-rew}\big(\tilde{y},f(\boldsymbol{x})\big)\Big] = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}(X,Y)}[l(y,f(\boldsymbol{x}))],$$

Define the optimal classifier f^* to minimize expected risk for underlying clean distribution and cross entropy loss as

$$f^* = \arg\min_f R_{\mathbb{P}(X,Y)}(f) = \arg\min_f \mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}(X,Y)}[l(y,f(\boldsymbol{x}))], \tag{18}$$

then $R_{\mathbb{P}(X,Y)}(f^*) = 0$ and for $l_{p-rew}\big(\tilde{y},f^*(\boldsymbol{x})\big) = 0$ for any $\boldsymbol{x}$. Note that the expected risk for posterior forward loss is non-negative and not larger than that for posterior reweight loss, we also have $l_{p-fw}\big(\tilde{y},f^*(\boldsymbol{x})\big) = 0$ for any $\boldsymbol{x}$. In addition, the difference of expected risk for two different classifiers satisfies

$$\begin{aligned} & \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-fw}\big(\tilde{y},f^*(\boldsymbol{x})\big)\Big] - \mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-fw}\big(\tilde{y},f(\boldsymbol{x})\big)\Big] \\ = \quad & -\mathbb{E}_{(\boldsymbol{x},\tilde{y})\sim\mathbb{P}(X,\tilde{Y})}\Big[l_{p-fw}\big(\tilde{y},f(\boldsymbol{x})\big)\Big] \leq 0, \end{aligned} \tag{19}$$

which implies that the classifier f^* can also minimize expected risk for underlying noisy distribution and posterior reweight loss. This completes the proof.

A.4 PROOF OF THEOREM 3.4

Aiming to standardize the constraint, we vectorize $W(\boldsymbol{x})$ as $c^2 \times 1$ dimension $\mathbf{w}_{vec}(\boldsymbol{x}) := [W_{11}(\boldsymbol{x}),\cdots,W_{c1}(\boldsymbol{x}),\cdots,W_{1c}(\boldsymbol{x}),\cdots,W_{cc}(\boldsymbol{x})]^\top$ and define the coefficient matrix $A(\boldsymbol{x})$ as follows,

$$A(\boldsymbol{x}) = \begin{bmatrix} \hat{\mathbb{P}}^\top(\tilde{\mathbf{Y}}|\boldsymbol{x}) & \cdots & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \hat{\mathbb{P}}^\top(\tilde{\mathbf{Y}}|\boldsymbol{x}) & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \hat{\mathbb{P}}^\top(\tilde{\mathbf{Y}}|\boldsymbol{x}) \end{bmatrix}_{c\times c^2} \tag{20}$$

where $\mathbb{P}^\top(\tilde{\mathbf{Y}}|\boldsymbol{x}) = [\mathbb{P}(\tilde{Y}=1|\boldsymbol{x}),\cdots,\mathbb{P}(\tilde{Y}=c|\boldsymbol{x})]_{1\times c}$. Therefore, the constraint $W^\top(\boldsymbol{x})\hat{\mathbb{P}}(\tilde{Y}|\boldsymbol{x}) = f(\boldsymbol{x})$ can be transformed as $A(\boldsymbol{x})\mathbf{w}_{vec}(\boldsymbol{x}) = f(\boldsymbol{x})$, which is a least-norm problem with undetermined equations constraint. Note that matrix $A(\boldsymbol{x})$ is full column rank, we can prove that the least-norm solutions is

$$\mathbf{w}^*_{vec}(\boldsymbol{x}) = A^\top(\boldsymbol{x})\Big(A(\boldsymbol{x})A^\top(\boldsymbol{x})\Big)^{-1}f(\boldsymbol{x}). \tag{21}$$

Considering any possible solution $\mathbf{w}_{vec}(\boldsymbol{x})$ satisfying the constraint $A(\boldsymbol{x})\mathbf{w}_{vec}(\boldsymbol{x}) = f(\boldsymbol{x})$, then $A(\boldsymbol{x})[\mathbf{w}_{vec}(\boldsymbol{x}) - \mathbf{w}^*_{vec}(\boldsymbol{x})] = \mathbf{0}$ and

$$\begin{aligned} [\mathbf{w}_{vec}(\boldsymbol{x}) - \mathbf{w}^*_{vec}(\boldsymbol{x})]^\top\mathbf{w}^*_{vec}(\boldsymbol{x}) &= [\mathbf{w}_{vec}(\boldsymbol{x}) - \mathbf{w}^*_{vec}(\boldsymbol{x})]^\top A^\top(\boldsymbol{x})\Big(A(\boldsymbol{x})A^\top(\boldsymbol{x})\Big)^{-1}f(\boldsymbol{x}) \\ &= \Big[A(\boldsymbol{x})[\mathbf{w}_{vec}(\boldsymbol{x}) - \mathbf{w}^*_{vec}(\boldsymbol{x})]\Big]^\top\Big(A(\boldsymbol{x})A^\top(\boldsymbol{x})\Big)^{-1}f(\boldsymbol{x}) = \mathbf{0}, \end{aligned}$$

which implies that the vectors $\mathbf{w}_{vec}(\boldsymbol{x}) - \mathbf{w}^*_{vec}(\boldsymbol{x})$ and $\mathbf{w}^*_{vec}(\boldsymbol{x})$ are perpendicular. Thus, the L_2 norm of $\mathbf{w}_{vec}(\boldsymbol{x})$ satisfies

$$\begin{aligned} ||\mathbf{w}_{vec}(\boldsymbol{x})||^2 &= ||\mathbf{w}^*_{vec}(\boldsymbol{x}) + \mathbf{w}_{vec}(\boldsymbol{x}) - \mathbf{w}^*_{vec}(\boldsymbol{x})||^2 \\ &= ||\mathbf{w}^*_{vec}(\boldsymbol{x})||^2 + ||\mathbf{w}_{vec}(\boldsymbol{x}) - \mathbf{w}^*_{vec}(\boldsymbol{x})||^2 \geq ||\mathbf{w}^*_{vec}(\boldsymbol{x})||^2. \end{aligned} \tag{22}$$

which shows that the equation (21) is the optimal solution. Via direct calculation, it is easy to obtain that $A(\boldsymbol{x})A^\top(\boldsymbol{x}) = \sum_{j=k}^c \hat{\mathbb{P}}^2(\tilde{Y}=k|\boldsymbol{x})\mathbf{I}_{c\times c}$. Therefore, the element of PTM satisfies

$$W^*_{ij}(\boldsymbol{x}) = \sum_{k=1}^c \hat{\mathbb{P}}^2(\tilde{Y}=k|\boldsymbol{x})\hat{\mathbb{P}}(\tilde{Y}=i|\boldsymbol{x})f_j(\boldsymbol{x}), \tag{23}$$

i.e., $W^*(\boldsymbol{x}) = \frac{f(\boldsymbol{x})\hat{\mathbb{P}}(\tilde{Y}|\boldsymbol{x})^\top}{||\hat{\mathbb{P}}(\tilde{Y}|\boldsymbol{x})||_2^2}$ and completes the proof.

A.5 PROOF OF THEOREM 3.5

Under this condition, independent reconstruction error implies that $Tr(e_{\hat{T}}(\boldsymbol{x})e_{\hat{W}}^{\top}(\boldsymbol{x})) = 0$. For Kalman transition matrix, the reconstruction deviation satisfies $W^*(\boldsymbol{x}) - W_{km}(\boldsymbol{x}) = (1-\lambda)e_T(\boldsymbol{x}) + \lambda e_W(\boldsymbol{x})$. Therefore, the reconstruction error is given by

$$\begin{aligned} \mathcal{L}_{mse}(\lambda) &= \mathbb{E}[\big((1-\lambda)e_T(\boldsymbol{x}) + \lambda e_W(\boldsymbol{x})\big)\big((1-\lambda)e_T(\boldsymbol{x}) + \lambda e_W(\boldsymbol{x})\big)^{\top}] \\ &= (1-\lambda)^2\mathbb{E}[\sigma_T(\boldsymbol{x})] + \lambda^2\mathbb{E}[\sigma_W(\boldsymbol{x})], \end{aligned} \tag{24}$$

It is easy to obtain that the optimal parameter $\lambda^* = \frac{\sigma_T(\boldsymbol{x})}{\sigma_T(\boldsymbol{x})+\sigma_W(\boldsymbol{x})}$ and the minimum reconstrution error $\mathcal{L}_{mse}(\lambda^*) = \frac{\sigma_T(\boldsymbol{x})\sigma_W(\boldsymbol{x})}{\sigma_T(\boldsymbol{x})+\sigma_W(\boldsymbol{x})} \leq \min(\sigma_T, \sigma_W)$.

B REPRODUCIBILITY

B.1 DATASET STATISTICS

For fairly comparison with previous work, we performance the image classification task on the three manually corrupted datasets, i.e., F-MNIST, SVHN, CIFAR-10, and one real-world noisy dataset Clothing1M. They have been widely adopted to study label noise problem. The detailed statistics are listed in Table 3.

Table 3: Dataset statistics on F-MNIST, SVHN, CIFAR-10, and Clothing1M

	F-MNIST	SVHN	CIFAR-10	Clothing1M
# training/noisy images	60,000	73,257	50,000	1,000,000
# test/clean images	10,000	26,032	10,000	10,000
label noise	synthetic	synthetic	synthetic	real-world

B.2 RUNNING ENVIRONMENT

All baselines and IF approaches are implemented in PyTorch, and tested on a machine with AMD EPYC 7282 16-core processors, 4 GeForce GTX-3090 Ti GPUs with 24GB memory size.

B.3 ALGORITHMS

We summarize the IF algorithm to correct the loss and provide the pseudo codes in Algorithm 1. For instance-dependent Label noise generation, we provide pseudo codes with controllable noise rate are provided in Algorithm 2. This algorithm follows the state-of-the-art method (Xia et al., 2020; Zhu et al., 2021a), where the overall noise rate is τ.

Algorithm 1: Information Fusion Algorithm

Input : Noisy dataset $\tilde{D} = (\boldsymbol{x}_n, \tilde{y}_n)_{n=1}^N$; Noisy validation data; tolerant epochs t.
Output : The robust neural network over noisy label.

Estimate the NTM $\hat{T}(\boldsymbol{x})$ according to (Xia et al., 2020) ;
while *Validation accuracy has increases in the last t epochs* **do**
  Calculate the PTM $\hat{W}(\boldsymbol{x})$ based on Equation (11) ;
  Obtain the uncertainty for noise and PTM ;
  Calculate the Kalman gain and Kalman transition matrix based on Equations (12) and (13) ;
  Correct the loss function as posterior forward loss with Kalman transition matrix $W_{km}(\boldsymbol{x})$;
end

Algorithm 2: Instance-Dependent Label Noise Generation

Input : Clean samples $(\boldsymbol{x}_n, y_n)_{n=1}^N$; Overall noise rate τ; Number of classes c; Size of input features: $1 \times d$.

Output : Noisy samples $(\boldsymbol{x}_i, \tilde{y}_n)_{n=1}^N$.

Sample flip rate q_i based on truncated normal distribution $\mathcal{N}(\tau, 0.1^2, [0, 1])$;

Sample the projection matrix $W \in \mathcal{R}^{d \times K}$ based on normal distribution $\mathcal{N}(0, 1^2)$;

for $n = 1$ *to* N **do**

  $p = \boldsymbol{x}_n W$ // Genrate instance dependent flip rate ;

  $p_{y_n} = -\infty$ // Control the diagonal elements of the transition matrix ;

  $p = q_n \cdot \text{softmax}(p)$ // Make the sum of the off-diagonal elements to be q_n ;

  $p_{y_n} = 1 - q_n$ // Set the diagonal element to be $1 - q_n$;

end

B.4 Learning Noise Transition Matrix via Decomposition

For the instance-dependent label noise, the NTM may be different for different samples. Thus, the complexity of NTM estimation is $O(Nc^2)$, which leads to ill-posed estimation problem. (Xia et al., 2020) introduces a part-dependent decomposition on the NTM, i.e., the noise of an instance depends only on its parts. Specifically, given L part-dependent transition matrix, e.g, $P^l \in [0, 1]^{c \times c}$, where $j = 1, \cdots, L$, the instance-dependent matrix can be approximated by a combination of part-dependent transition matrix and the combination coefficients, defined as $\boldsymbol{h}(\boldsymbol{x}) \in [0, 1]^L$, depend on the feature vectors. Then the instance-dependent transition matrix $W(\boldsymbol{x})$ can be approximated as follows

$$\hat{T}(\boldsymbol{x}) = \sum_{l=1}^{L} \boldsymbol{h}_j(\boldsymbol{x}) P^l. \tag{25}$$

where $\boldsymbol{h}_j(\boldsymbol{x})$ represents the j-th elements of $\boldsymbol{h}(\boldsymbol{x})$ and satisfies the normalization condition $||\boldsymbol{h}(\boldsymbol{x})|| = 1$ for any features vector to maintain that the summation of column is 1 for NTM.

In this work, we follow (Xia et al., 2020) to learn the NTM in three steps: (i) learning combination coefficients $\boldsymbol{h}(\boldsymbol{x})$ via parts-based features representation (Lee & Seung, 1999; Tao et al., 2017); (ii) learning the instance-dependent transition matrix by exploiting anchor points (Liu & Tao, 2015; Xia et al., 2019); (iii) learning the part-dependent transition matrices via minimization of reconstruction error for instance-dependent transition matrix.

(Xia et al., 2020) justifies the part-dependent decomposition assumption via parts-based representations. For example, (Hosseini-Asl et al., 2016) shows that the input features rely on parts-based representations in object recognition. Thus, it is natural to approximate the label noise in the part level. In display advertising, the delay feedback of the label (Ktena et al., 2019; Yasui et al., 2020) introduces label noise. Earlier the data is generated, the less probability the feedback exists. Thus, the label noise highly depends on part of features.

Since the instance-dependent NTM may be poorly learned, the slack variable trick (Xia et al., 2019; 2020) is also adopted to modify the instance-dependent transition matrix, which significantly improve the accuracy. However, the slack variable trick ignores the posterior information for each instance, which limits the accuracy. In addition, the variance of accuracy in (Xia et al., 2020) is large since there is not any guide information to modify the instance-dependent NTM except minimization of the corrected loss. In this paper, we claim that the measurement information is useful to modify the NTM. We will introduce how to estimate the PTM and information fusion in the next two subsections.

B.5 Implementation Details

Warm-up Training Strategy Aiming to estimate the transition matrix, we adopt warm-up training strategy that training the neural network via cross entropy loss with noisy data. we use SGD as the optimizer with 0.9 momentum, 128 batch size, and 10 epochs. With the benefit of memorization effect (Arpit et al., 2017), warm-up training with appropriate epochs can guarantee that the predicted probability of neural network is reliable and can approximate underlying clean probability.

Training with Corrected Loss Our proposed PTM estimation relies on the neural network predicted probability. Although the warm-up training can guarantee the prediction accuracy for clean instances with the benefits of memorization effect, the prediction of the PTM estimation may still be not reliable. To ease the dilemma, we adopt sample selection to choose the reliable prediction and thus guarantee the reliable PTM estimation for specific instance. Similar to (Cheng et al., 2021), the small-loss criteria is adopted to select the instances with small training loss for training. Specifically, we select the instance via the weighted loss if the instance loss is less than the average loss over all classes.

For the instance satisfying the small loss criterion, we have confidence that the instance is clean, which means the PTM is identity. To put it simply, we sum the estimated PTM and identity and then adopt column normalization. It is worth noting that the noisy labels are adopted in forward propagation, which is highly different with previous transition matrix-based works (Xia et al., 2019; Liu & Tao, 2015; Patrini et al., 2017; Xia et al., 2020). Compared with hard sample selection based methods (Cheng et al., 2021), our method achieves the soft sample selection and adaptively correct the loss for “clean" instance.

C Additional Experimental Results

C.1 Stability on Synthetic Noisy Datasets

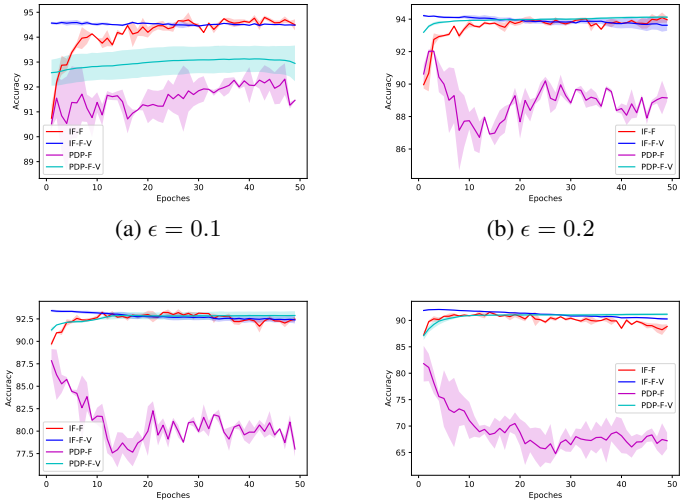
We study the training stability via inspecting the test accuracy over epochs. Figures 5 and 6 show the test accuracy with respect to epochs on SVHN and F-MNIST datasets with noise rate IDN-10% to IDN-40%. It is seen that our proposed method IF can achieve better accuracy and stable training compare with baseline PTD, which implies that the PTM fusion can decrease the transition matrix estimation error and improve the training stability. T-Revision trick (-V method) proposed in (Xia et al., 2019) aims to compensate the NTM and thus improve the accuracy performance. It is seen that PTD-F-V can achieve better performance than PTD-F, which verifies that the learned NTM is not very accurate. For our proposed method, IF-V can achieve comparable performance than IF-F and demonstrates that transition matrix revision trick can not improve the performance since PTM has already been employed to compensate the transition matrix error.

Compared with different noise rates, it is seen that the accuracy of PTD method may decrease with respect to epochs, especially for large noise rate. Under large noise rate, the number of clean samples is limited and the neural network may already overfit noisy samples during warm-up training and thus hurt the transition matrix estimation. In this scenario, PTD method does not have the capability to adjust such poor NTM estimation and thus leads to the accuracy drop during training. Our proposed method IF and T-Revision trick both adjust the estimated transition matrix and mitigate the accuracy drop.

D Related work

Label Noise Model There are three types of label noise model, including the random classification noise (RCN) model (Manwani & Sastry, 2013; Natarajan et al., 2013), the class-conditional label noise (CCN) model (Patrini et al., 2017; Xia et al., 2019; Zhang & Sabuncu, 2018; Liu & Guo, 2020) and the e instance-dependent label noise (IDN) model (Cheng et al., 2020; Xia et al., 2020; Cheng et al., 2021; Berthon et al., 2021; Zhu et al., 2021a). Specifically, RCN means that each label is flipped independently with a constant probability, CCN assumes that the flip probability (noise rates) are the same for the instance with the same clean labels. IDN is the most general case, where the flip probability depends on its instance.

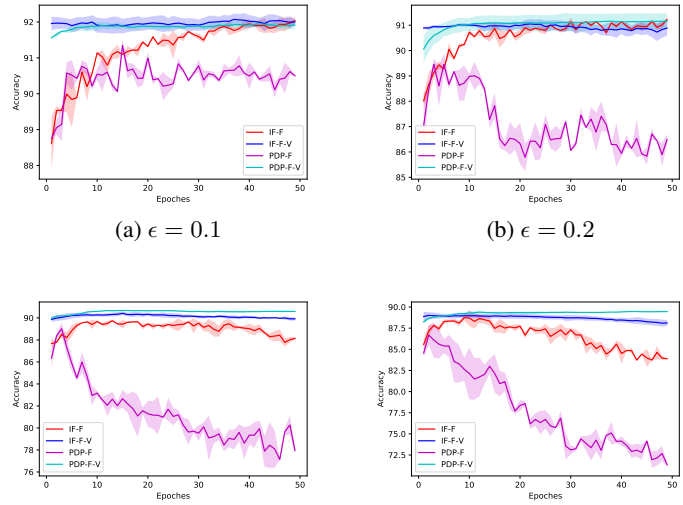
Loss Correction via Transition Matrix The transition matrix bridges the gap between the model predicted probability for noisy and clean data, which can achieve *risk-consistent* classifier with label noise. The transition matrix can be estimated via cross-validation method (Natarajan et al., 2013), anchor points assumption (Xia et al., 2019; Yu et al., 2018; Yao et al., 2020). Subsequently, the estimated transition matrix can be adopted to correct the loss function via forward correction (Patrini et al., 2017), backward correction (Patrini et al., 2017) and reweight (Xia et al., 2019; Liu & Tao, 2015).



(a) $\epsilon = 0.1$ (b) $\epsilon = 0.2$

(c) $\epsilon = 0.3$ (d) $\epsilon = 0.4$

Figure 5: The test accuracy on SVHN datasets with different levels of IDN noise.



(a) $\epsilon = 0.1$ (b) $\epsilon = 0.2$

(c) $\epsilon = 0.3$ (d) $\epsilon = 0.4$

Figure 6: The test accuracy on F-MNIST datasets with different levels of IDN noise.

Loss Correction via Sample Selection Due to the experiments observation– the neural network learns clean instances first (Arpit et al., 2017) and memorizes the instances gradually, there are lots of methods attempting to identify mislabeled training examples and then filter them out (Brodley & Friedl, 1999; Jiang et al., 2018; Han et al., 2018b; Yu et al., 2019; Angelova et al., 2005; Huang et al., 2019; Li et al., 2019). The core idea is to select the “small loss" samples as clean one.