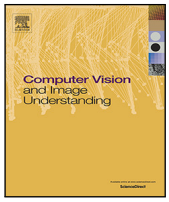
Computer Vision and Image Understanding 235 (2023) 103771

Contents lists available at ScienceDirect

Computer Vision and Image Understanding

journal homepage: www.elsevier.com/locate/cviu



GEIKD: Self-knowledge distillation based on gated ensemble networks and influences-based label noise removal

Fuchang Liu [a], Yu Wang [b], Zheng Li [c], Zhigeng Pan [d,*]

[a] School of Information Science and Technology, Hangzhou Normal University, China
[b] Alibaba Business School, Hangzhou Normal University, China
[c] College of Computer Science, Nankai University, China
[d] School of Artificial Intelligence/ School of Future Technology, Nanjing University of Information Science and Technology, China

ARTICLE INFO

Communicated by Nikos Paragios

Keywords:
Self-distillation
Gated ensemble network
Influences estimation
Noisy labels

ABSTRACT

Self-distillation has gained widespread attention in recent years because it progressively transfers the knowledge in end-to-end training schemes within one network. However, self-distillation methods are susceptible to label noise hence leading to poor generalization performance. To address this problem, this paper proposes a novel self-distillation method, called *GEIKD*, which combines a gated ensemble self-teacher network and the influences-based label noise removal. Specifically, we design a gated ensemble self-teacher network composed of multiple teacher branches, which allows a gated fused knowledge based on a weighted bi-directional feature pyramid network. Moreover, we introduce influences estimation into the distillation process to quantify the effect of noisy labels on the distillation loss, and then reject the unfavorable instances as noisy labeled samples according to the calculated influences. Our influences-based label noise removal can be integrated with any existing knowledge distillation training schemes. The impact of noisy labels on knowledge distillation can be significantly alleviated by the proposed noisy instances removal with little extra training efforts. Experiments show that the proposed *GEIKD* method outperforms the state-of-the-art methods on CIFAR-100, TinyimageNet and fine-grained datasets CUB200, MIT-67, Stanford40 and FERC dataset, using clean data and data with noisy labels.

1. Introduction

Deep learning has now achieved great success in diverse fields including image recognition, speech recognition and natural language processing. However, many vision applications require to deploy lightweight networks on the edge devices with limited memory and computational capacity. Various model compression techniques have been proposed to overcome this challenge including low-rank approximation (Denil et al., 2013; Novikov et al., 2015), pruning (Han et al., 2015a,b), quantization (Courbariaux et al., 2015; Rastegari et al., 2016), compact network design (Iandola et al., 2016; Howard et al., 2017) and knowledge distillation (KD) (Hinton et al., 2015). Knowledge distillation has been received much attention since it is much easier to deploy without significant loss in performance.

Originally proposed by Hinton et al. (2015), knowledge distillation aims to transfer the knowledge from a large unwieldy teacher model to a single smaller student model. Knowledge distillation has gained prominence for practical real-world applications. Typical knowledge distillation uses the logits or activations of intermediate layers or the relationship between activations and feature maps as the source of teacher's knowledge. The hypothesis is that a small student model will learn to mimic a large teacher model and utilize the knowledge of the teacher to achieve similar or even higher accuracy. It assumes that the teacher model should be more powerful than the student model. However, this is not necessarily true. Offline knowledge distillation uses large pre-trained networks to guide the student network, thus requires a large capacity deep neural network as the teacher network which may not be available for real use cases. To address this limitation, online knowledge distillation is proposed, leveraging a set of peer students as the teacher network (Zhu et al., 2018; Li et al., 2020a; Chen et al., 2020). In contrast to traditional offline knowledge distillation, online knowledge distillation gives competitive distillation performance and higher computational efficiency.

Self-knowledge distillation is another principal method which uses the same network for the teacher and the student network. It is mainly divided into two types: data augmentation-based and auxiliary network-based approaches. Data augmentation-based self-knowledge distillation learns consistent predictions of synthetic training data through different augmented versions of a single instance or a pair of instances from the same class (Xu and Liu, 2019; Yun et al., 2020). Auxiliary network-based approaches utilize assistive branches (Zhu

* Corresponding author.
E-mail address: zgpan@nuist.edu.cn (Z. Pan).

https://doi.org/10.1016/j.cviu.2023.103771
Received 13 March 2023; Received in revised form 22 May 2023; Accepted 30 June 2023
Available online 10 July 2023

et al., 2018; Chen et al., 2020; Zhang et al., 2019) to mimic outputs for each other so that it is beneficial to knowledge transferring and weight regularization. There are some other networks designed to use historical information to provide additional supervision information (Yuan et al., 2020; Kim et al., 2021), like logits or feature maps from earlier epochs of the teacher model.

Noisy labels have also received widespread attention with in-depth study of deep learning (Nguyen et al., 2019; Natarajan et al., 2013). Neural networks in many practical scenarios heavily depend on access to high-quality labeled training data. Label noise in large training datasets is ubiquitous due to measure errors (lack of professional expertise or carelessness). Studies have shown that label noise in training data can greatly reduce the accuracy of deep neural networks on clean test data. Especially for supervised image classification tasks, the classical loss function such as cross-entropy (CE) is not noise-tolerant. If training datasets contain examples with noisy labels, models are prone to overfitting and less generalizable. Similarly, in knowledge distillation applications, we observe that many self-KD methods also degrade their performance due to the presence of label errors in training data. Unfortunately, few works on KD attempt to understand noisy training labels, as well as mitigation strategies, to overcome them.

Another observation is that most of current self-distillation approaches usually use the same network for the teacher and the student models without guaranteeing the superiority of teacher networks. To address above issues, we propose a gated fusion-based self-distillation architecture to improve the generalization ability of teachers. The proposed method transfers distinct kinds of knowledge to the target network by introducing an assistive auxiliary self-teacher network. Furthermore, the knowledge form multiple teachers is fused as the average response across all branches. Compared with other self-knowledge distillation, our method achieves the overwhelming performance in image classification tasks on various datasets and shows high noise-tolerance. This result reflects that auxiliary network-based methods are more stable to label noise than data augmentation-based self-knowledge distillation methods.

Despite achieving good performance on data with noisy labels, the proposed self-distillation method is unable to directly eliminate label noise, like most of KD methods. In other words, we only passively mitigate the impact of noisy labels, and lack an effective method to actively remove noisy labels or even understand noisy labels by studying their impact of noise level. To this end, we introduce an influences estimation into KD framework to quantify the impact of noisy labeled instance on distillation loss and remove unfavorable instances with negative influences. The idea behind this is to directly distinguish and reject noisy instances using their influences on the training loss. Specifically, the influence function is plugged into the self-KD training process, and gives the influence of every training sample on the fly. Subsequently, these samples with significantly negative impact on the target network are regarded as samples with incorrect labels, so that these samples are removed and their impact is mitigated simultaneously.

Our main contributions can be summarized as follows.

1. We propose a gated ensemble self-teacher network consists of multiple self-teacher branches, which provides fused knowledge from multi-resolution feature maps through a learnable gated fusion, thereby further optimizing knowledge transfer between the teacher model and the student model. Our self-teacher architecture outperforms the state-of-the-art methods in different classification tasks and shows the robustness of self-distillation against noisy labels without any data augmentation.
2. To alleviate performance degradation caused by noisy labels, we propose an influences-based label noise removal via plugging the influences estimation into the self-KD training process, which dynamically removes samples labeled incorrectly from the training set using their calculated influences during the training process. To the best of our knowledge, we are the first method to actively remove label noise in the self-distillation process by exploiting the influences of samples on the distillation loss. It is empirically demonstrated that the proposed label noise removing method is effective and able to combine with any existing knowledge distillation network in a plug-and-play mode.
3. Extensive experiments show that the proposed distillation approach outperforms the state-of-the-art methods against different noise level. Moreover, t-SNE visualization results show that noisy samples are identified precisely and knowledge learned from our student–teacher network are more discriminative.

2. Related work

Knowledge distillation. Knowledge distillation is the process of transferring the knowledge from a large complex network (the teacher network) to a simpler network (the student network) by training the student to mimic the teacher's output or transferring crosssample relations from the teacher to the student. The different kinds of knowledge are categorized into three types: response-based knowledge (Hinton et al., 2015; Li et al., 2023), feature-based knowledge (Chung et al., 2020; Heo et al., 2019; Romero et al., 2014), or relation-based knowledge (Tian et al., 2020; Liu et al., 2019; Park et al., 2019). Hinton et al. formalized the knowledge distillation framework which uses the student network to match the logits of the teacher network (Hinton et al., 2015). The follow-up works on distilling knowledge using logits are called response-based knowledge. Romero et al. proposed FitNet based on implicit learning, which distills knowledge by minimizing the distance of feature maps between the student and the teacher (Romero et al., 2014). Inspired by FitNet, the following works (Chung et al., 2020; Heo et al., 2019) capture knowledge of intermediate layers and train the student model to learn the same feature activations as the teacher model, namely feature-based knowledge. In addition to knowledge represented in logits and the intermediate layers, knowledge that captures the relationship between feature maps can also be used to train a student network, which is called relation-based knowledge (Tian et al., 2020; Liu et al., 2019; Park et al., 2019). Methods which falls into this category are called offline distillation and share the similarity of using pre-training of complex teacher models to improve the generalization ability of student networks by knowledge migration. Despite the robustness of offline methods, very recent research trends are leaning toward online approaches since offline distillation requires additional training time and computational cost. Online distillation utilizes a group of students to learn from each other, which leads to performance improvement and easier deployment.

Self-knowledge distillation. Self-knowledge distillation uses the same network for the teacher and the student models to avoid training a large teacher network. The recent works on self-knowledge distillation is largely divided into data augmentation based approaches and auxiliary network based approaches. DDGSD (Xu and Liu, 2019) improves the generalization ability of student networks by forcing networks to predict consistently across these distorted versions of data. The idea behind DDGSD is to transfer knowledge between different distorted versions of the same training data via data augmentation. CS-KD (Yun et al., 2020) transfer knowledge by matching predictive distributions between different samples of the same label during training, which can be categorized as relation-based knowledge distillation. BYOT (Zhang et al., 2019) uses a set of shallow networks with classifiers as students and train them via distillation from the deepest one. ONE (Zhu et al., 2018) presented an one-stage online distillation based on the teacher model building on the ensemble of multiple auxiliary branches which share the same low-level layers. FRSKD (Ji et al., 2021) utilizes an auxiliary self-teacher network to transfer a refined knowledge for the classifier network.

As mentioned above, BYOT and ONE and FRSKD belong to auxiliary network based approaches. It is worth noting that: these methods based on auxiliary networks give the way of designing a strong teacher using

Table 1
Comparison of state-of-the-art noisy label combating techniques with our influence-based label noise removing approach. As shown in the first column, a comparison in terms of five index is given to investigate cons and pros. "large class": the ability of tackling a large number of classes; "heavy noise": the ability of combating with high level of noise; "flexibility": the ability of integrating into an existing network architecture; "no pre-train": pre-train requirement or trained from scratch; "plug into&out": plug-and-play or not applicable.

	Bootstrap (Reed et al., 2014)	S-model (Goldberger and Ben-Reuven, 2017)	F-correction (Patrini et al., 2017)	Decoupling (Malach and Shalev-Shwartz, 2017)	MentorNet (Jiang et al., 2018)	Co-teaching (Han et al., 2018)	SELF (Nguyen et al., 2019)	DivideMix (Li et al., 2020b)	LongReMix (Cordeiro et al., 2023)	Influences-based removal
large class	✗	✗	✗	√	√	√	√	√	√	√
heavy noise	✗	✗	✗	✗	√	√	√	√	√	√
flexibility	✗	✗	√	√	√	√	√	√	√	√
no pre-train	√	✗	✗	✗	√	√	√	√	√	√
plug into&out	✗	✗	✗	✗	✗	✗	√	√	√	√

auxiliary branches but no guarantee of the generalization ability of the auxiliary networks, which is a critical limitation of the student networks to learn knowledge from the teacher model. To overcome this limitation, our method develops an elegant auxiliary network to transfer generalization ability from a gated ensemble self-teacher network to the student network effectively. Consequently, the purpose of our auxiliary self-distillation network is to generate a strong teacher which can acquire knowledge from multi-resolution feature maps.

Noise label. Noise is ubiquitous in training and inference. Noisy labels is a kind of annotation errors and introduce an aleatoric uncertainty into the network. It has been demonstrated that the classification accuracy of networks using training data contains samples with noisy labels deteriorates significantly compared to the counterpart on clean training data. Therefore, the impact of noise on classification networks cannot be ignored or evaded. To tackle noisy labels, Reed et al. (2014) proposes a bootstrap technique for correcting labels in real-time to mitigate the potential influences caused by noisy labels. S-model (Goldberger and Ben-Reuven, 2017) alleviates the influences of noisy labels by an additional softmax layer modeling the noise transfer matrix. Similarly, F-correction (Patrini et al., 2017) corrects the prediction by the noise transfer matrix for noise suppression. The author states that F-correction is the first work to train a standard network for estimating the transfer matrix. Decoupling (Malach and Shalev-Shwartz, 2017) suppresses noise by updating the model parameters only using samples with inconsistent prediction from two classifiers. MentorNet (Jiang et al., 2018) adopts a pre-training network as an assistive teacher, then uses it to filter out noisy instances. This method allows its student network to learn robustly under noisy labels. Co-teaching (Han et al., 2018) proposes a new deep learning paradigm, called "co-teaching", to combat noisy labels. Self-integrating label filtering (SELF) (Nguyen et al., 2019) progressively filters out incorrect labels during the training process. SELF improves the performance of task networks by gradually allowing supervision from only potential non-noisy (clean) labels and blocking learning from filtered noisy labels. DivideMix (Li et al., 2020b) divides the training data into two sets in a semi-supervised manner: one with presumably clean labels and the other with noisy labels, and then trains two separate models on these two datasets. DivideMix can effectively learn from both clean and noisy labels, thus improving the model's robustness and performance on noisy datasets. LongReMix (Cordeiro et al., 2023) is a novel noisy-label learning algorithm that initially applies unsupervised learning to separate clean from noisy training samples, then utilizes a semi-supervised learning stage to minimize the empirical vicinal risk by using a set of clean-labeled samples and an unlabeled set of noisy-classified samples. LFBS (Baek et al., 2022) proposes a self-distillation framework that leverages the predictions from existing learning with noisy labels (LNL) models. This framework aims to enhance performance through a process of rectified distillation, utilizing both hard pseudo labels and feature distillation. Han et al. (2019) propose an efficient and effective iterative learning structure, Self-Learning with Multi-Prototypes, which is used to reassign labels to noisy samples and to train a convolutional neural network (ConvNet) on a truly noisy dataset, without requiring extra clean supervision. NLKD (Liang et al., 2021) introduces a knowledge distillation-based noise learning framework, which is designed to enhance segmentation outcomes on impure data. The framework employs a teacher network to guide the student network through the knowledge distillation phase. Both the teacher and student create pseudo-labels and jointly evaluate the quality of annotations to generate weights for each sample.

Table 1 provides a systematical comparison of the aforementioned methods. The table clearly shows that our influences-based label noise removal does not depend on any specific network architecture and allows to handle datasets with a large number of categories and high noise level, and also supports training from scratch. Furthermore, influences-based label noise removal provides a plug-and-play mode for any existing networks and can be carried out on the fly during the distillation process. These advantages make the proposed influences-based label noise removal more attractive for a wide range of real-world application scenarios, such as poisoning attacks with mislabeling. We has some similarity with SELF on the use of self-ensemble, but for different purposes. SELF and other methods above are targeted to semi-supervised learning, while our method is dedicated to the KD problem. We have carefully studied the SELF method and found that it is not straightforward to apply it to the KD problem. The reason is that SELF relies on a Mean-Teacher model to reject label noise, which is time-consuming to train in the online distillation framework. Specifically, it is impossible to flip the label for every sample to cover all the label classes in a brute-force way and then to update the Mean-Teacher model to obtain a better model. Such distillation training process is more violent and not applicable to dataset with a large number of classes. In contrast, our approach identifies the possible samples with noisy labels by the theoretically guaranteed influences estimation and then removes them directly without any violent attempts. The proposed influences-based label noise removal allows the performance of the distillation network to be restored to the state-of-the-art level at the price of a slight increment of training time.

3. Method

In this section, we describe our gated ensemble network and influe nces-based label noise removal. Fig. 1 shows the pipeline of our distillation method. The proposed gated integrated self-distillation method is introduced in Section 3.1. Moreover, the specific details of the influences estimation are given in Section 3.2.

3.1. Gated ensemble self-teacher network

The main purpose of this network is to design a student–teacher network architecture which optimizes knowledge transfer via gated multi-resolution features fusion structures. The proposed teacher network incorporates BiFPN (Tan et al., 2020) architecture into the ensemble of multiple auxiliary branches. Specifically, the top-down and bottom-up fusion of BiFPN are both used to generate refined feature maps for the teacher. Finally, an appealing teacher network can be achieved to transfer knowledge via the soft targets based on fused logits.

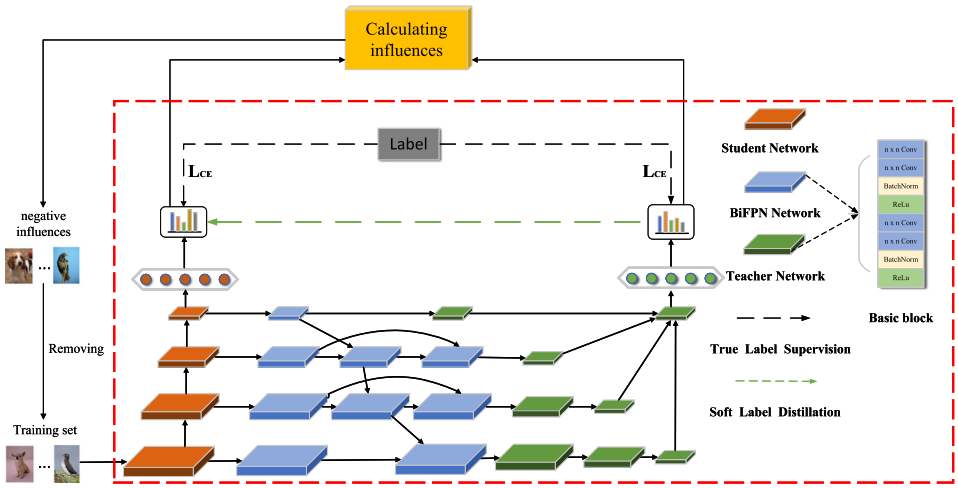


Fig. 1. Overview of the *GEIKD* framework. This framework is composed of a gated ensemble self-teacher network (inside the red dashed box) and the KD training scheme using influences-based label noise removal (outside the red dashed box). The first part aggregates features of different resolutions through top-down fusion and bottom-up fusion and produces a prediction based on fused logits as a gated fusion of knowledge from different branches. The second part removes label noise by calculating the impact of each sample on training loss via influences estimation. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

As illustrated in Fig. 1, the blue blocks represents BiFPN modules which fuse features by repeatedly applying top-down and bottom-up bidirectional feature fusion. Branch networks and the target network share the same backbone with convolution blocks consist of convolutional, batch normalization and Relu layers. Feature maps with different resolutions are downsampled progressively. Knowledge given by different branches are aggregated to produce a refined feature maps through a learnable gated fusion. Moreover, soft targets are utilized to further assist the knowledge distillation of the target network.

We formulate our network loss based on architectures described above

$$L = L_{ce}^{S}(x, y; \theta_s) + f_{fuse}(x, \theta_t) + \lambda L_{KD}(x; \theta_s, \theta_t) \tag{1}$$

$L_{ce}^{S}(x, y; \theta)$ is the student network cross-entropy loss function, f is the classifier network, and θ is the parameter of the classifier network. λ is the hyperparameter which controls the trade-off between the KD loss term with other loss terms. The student network and the ensemble of teacher networks are all trained in supervised way using the cross-entropy loss.

In our network, each branch is considered as a separate teacher, finally the average of all branches can generate a more powerful teacher. We employ a learnable gated fusion to implement the fusion of knowledge from all branches, which is written as follows

$$f_{fuse}(x, \theta_t) = \sum_{i}^{m} g_i . f_i^t(x, \theta_t) \tag{2}$$

where m is the number of branches in the self-teacher network and θ_t is the parameter of self-teacher network. f_i^t denotes the network for branch i, g_i is the learnable weight for branch i, *i.e.*, the gating variable acts as a gatekeeper, controlling the weight of each branch (Zhu et al., 2018). To improve generalization capabilities, vanilla knowledge distillation transfers the teacher's knowledge to the student network by minimizing the Kullback–Leibler (KL) loss between the teacher and the student model. We also follow the same loss metric to define the distillation loss function

$$L_{KD}(x; \theta_s, \theta_t) = D_{KL}(softmax(\frac{f_s(x; \theta_s)}{T}) \parallel softmax(\frac{f_{fuse}(x; \theta_t)}{T})) \tag{3}$$

where T is the temperature scaling parameter and θ_s gives the parameters of the student network.

Self-knowledge distillation distills the knowledge in a model itself and can be considered a special case of online distillation. Hence, it can be interpreted within an online framework of knowledge distillation as the student and the teacher are identical. In our gated ensemble self-teacher network, we train the target network as the student network and the auxiliary network generated by the target network as the teacher network. The target network will become better by mimicking predictions of the ensemble self-teacher networks.

3.2. Influences-based self-knowledge distillation

Our simple rationale: instances that are noisy labeled can negatively affect training loss and should therefore be eliminated. This rationale also extends to self-distillation frameworks. Similar to conventional training, noisy labeled instances can harm the teacher network's ability to generalize. To alleviate the impact of noisy labels, we integrate an influence estimation method into the self-knowledge distillation framework.

We reformulate distillation networks using an ensemble of auxiliary networks as follows

$$\hat{\theta} := \arg\min_{\theta \subseteq \Theta} \frac{1}{n} \sum_{i=1}^{n} L_i(\theta) \tag{4}$$

where $L_i(\theta)$ denotes the loss function of either the teacher network $L_{ce}^{t_i}(x, y; \theta_t)$ or the student network $L_{ce}^{S}(x, y; \theta_s)$.

However, to discover the instance with negative influence on the final loss is always required to retrain the network after removing a sample and thus prohibitively slow. To alleviate the computational overhead of retraining the network, a small parameter ϵ is used to approximate the change of the model's prediction after removing a training sample by setting $\epsilon = -\frac{1}{n}$. The model parameters of the student and teacher networks for removing a training sample on the training loss at a clean test instance are estimated by solving the following empirical risk minimization

$$\tilde{\theta} := \arg\min_{\theta \subseteq \Theta} \frac{1}{n} \sum_{i=1}^{n} L_i(\theta) + \epsilon L(\theta) \tag{5}$$

With the parameters derived from Eq. (5), we can evaluate the change of the loss after removing a training sample on the validation set as follows (namely influences)

$$\delta(\epsilon) = L^{v}(\tilde{\theta}) - L^{v}(\hat{\theta}) \tag{6}$$

If $\delta(\epsilon) > 0$ and $\epsilon = -\frac{1}{n}$, then $L^{v}(\tilde{\theta}) - L^{v}(\hat{\theta}) > 0$, which indicates that the loss on the validation set is increased after removing this training instance. In other words, $\delta(\epsilon) > 0$ means that the ith training sample is beneficial for network classification. Koh and Liang (2017) (Koh and

Liang, 2017) develop the linear approximation calculation of the influence function $\delta(\epsilon) \simeq \epsilon \Phi(x_i, x_v, \hat{\theta})$, where $\Phi(x_i, x_v, \hat{\theta})$ has a closed-form expression that can be derived as

$$\Phi(x_i, x_v, \hat{\theta}) = -\nabla_\theta L^v(x^v, \hat{\theta})^T H_\theta^{-1} \nabla_\theta L(x_i, \hat{\theta}) \tag{7}$$

where $H_\theta = \frac{1}{n} \sum_{i=1}^{n} \nabla_\theta^2 L(x_i, \hat{\theta})$ is the Hessian matrix of the loss function for the training samples. We employ the stochastic estimation procedure of Koh and Liang (2017) to simplify the computation of the product between the inverse Hessian matrix and the gradient vector relying on the HVP Pearlmutter (1994) method. Specifically, for any training instance, we can use $\nabla_\theta L(x, \hat{\theta})$ as the unbiased estimator of H. This gives us the following procedure to uniformly sample the sampled points from the training data. Define $H_0^{-1} v = v$, and compute cyclically

$$H_j^{-1} v = v + (I - \nabla_\theta^2 L(\hat{\theta})) H_{j-1}^{-1} v \tag{8}$$

If $\epsilon = -\frac{1}{n} < 0$ and $\Phi(x, x_v, \hat{\theta}) < 0$, thus $\delta(\epsilon) > 0$. That is, the sample x_i is beneficial for network classification. In our distillation training, the effect of x_i on the validation sets of the student network and the teacher network can be calculated as:

$$\begin{aligned} \Phi_i = \Phi_i^T + \Phi_i^S = & \sum_{j=1}^{m} -\nabla_\theta L_j^v(x^v, \hat{\theta}_S)^T H_\theta^{-1} \nabla_\theta L_j(x_i, \hat{\theta_S}) \\ & + \sum_{j=1}^{m} -\nabla_\theta L_j^v(x^v, \hat{\theta}_T)^T H_\theta^{-1} \nabla_\theta L_j(x_i, \hat{\theta_T}) \end{aligned} \tag{9}$$

As previously stated, $\Phi_i < 0$ and $\epsilon = -\frac{1}{n} < 0$ means that $\delta(\epsilon) > 0$, which means x_i is beneficial on average for knowledge distillation training. Moreover, the smaller Φ_i is, the more likely x_i is to be a favorable sample, and vice versa. Thus, Φ_i can be used as a reasonable quantitative indicator to identify which training sample is beneficial to the performance of the classifier.

The entire process of influences estimation and label noise removal is summarized in Algorithm 1. We compute the influence for each sample and rank training samples ascendingly according to their influences, finally remove the top k elements or elements with negative influences as label noise from training datasets.

Algorithm 1: Influences-based label noise removal: Given training set D_{tr}, validation set D_{va}, categories N, samples in each category M, training rounds L, rejected samples k, regularization constant λ.

```
if epoch < L then
    for class i < N do
        compute Γ = ∇_θ L^v(x_i^v, θ̂)^T H_θ^{-1}
        for sample j < M do
            compute δ_{i,j}(ϵ) = (1/m) Σ_{j=1}^{m} Γ ∇_θ L(x_{i,j}, θ̂)
            I ← {x_{i,j}, δ_{i,j}}
        end
    end
    x*, δ* ← TopN(Rank(I.δ), k)
    remove x_{i,j} ∈ x* from D_tr
end
```

4. Experiments

4.1. Datasets and settings

Datasets. To demonstrate the effectiveness of our approach, we selected five commonly used public datasets: CIFAR-100 (Krizhevsky et al., 2009), TinyImageNet (Krizhevsky et al., 2009) and Facial Expression Recognition Challenge[1] (FERC), Caltech-UCSD Bird (CUB200) (Wah et al., 2011), MIT Indoor Scene Recognition (MIT-67) (Quattoni and Torralba, 2009) and Stanford40 Actions (Stanford40) (Yao et al., 2011). To make a unified size for training images, we resize the images from TinyImageNet to the size of 32 × 32 as CIFAR-100. Different from CIFAR-100 and TinyImageNet, CUB200, MIT-67, Stanford40 are used for the fine-grained visual recognition (FGVR) task, which contain fewer instances per class.

Setup. We perform the classification test on FERC,CIFAR-100 and TinyImageNet using ResNet18 (He et al., 2016) and WRN-16-2 (Zagoruyko and Komodakis, 2016) as backbones. To adapt ResNet18 to small datasets, we modified the first convolutional layer of ResNet18 with a 3 × 3 kernel size, a single stride, and a single padding, and without the max-pooling operation. We keep the standard ResNet18 for the FGVR task test. For all classification experiments, we employ stochastic gradient descent (SGD) with an initial learning rate of 0.1 and a weight decay of 0.0001. We set the total epoch to be 200 and the batch size to be 128. Especially for the experiments on CIFAR-100, TinyImageNet and FERC, we reduce the learning rate by 10 times when the training epochs are equal to 100 and 150, respectively. We use the standard data augmentation in all experiments methods, i.e., random cropping and flipping. In regard to hyperparameters, we set λ to be 3 for CIFAR-100 and 4 for TinyImageNet. In addition, we set the temperature parameter T to 4. We use the noise label method in the previous paper (Han et al., 2018; Nguyen et al., 2019; Li et al., 2020b; Cordeiro et al., 2023; Wang et al., 2019; Baek et al., 2022) to set the symmetric and asymmetric noise label methods. We consider the typical situation with symmetric and asymmetric label noise. In the case of the symmetric noise, a label is randomly flipped to another class with probability p. Asymmetric noise is designed to replaced by similar classes.

Compared methods. We compare some representative methods. DDGSD (Xu and Liu, 2019) is an advanced data augmentation based distillation method which forces networks to make consistent predictions across distorted data. CS-KD (Yun et al., 2020) forces the student and the teacher to output consistent predictive distribution between different samples from the same label. BYOT (Zhang et al., 2019) and ONE (Zhu et al., 2018) and FRSKD (Ji et al., 2021) use auxiliary networks as self-teachers to transfer knowledge to the target network. We run these models using their release code and apply the same training settings and hyperparameters settings as mentioned above. We use resnet18 and WRN-16-2 as the baseline networks.

4.2. Evaluation results of gated ensemble self-teacher network

In this subsection, we evaluate the baseline methods and the proposed ensemble networks on the common image classification datasets and FGVR datasets.

Comparison to the state-of-the-art. As shown in Table 2, our method yields better performance compared to other existing methods on clean data. Specifically, our method improves the classification accuracy by 3.85% ∼ 4.54% on CIFAR-100 and 1.97% ∼ 5.10% on TinyImageNet dataset compared to baselines. Compared to the state-of-the-art approaches, our results achieve the highest classification accuracy. We also conducted experiments on the common datasets on FGVR tasks. As shown in Table 3, Our method still gives a competitive classification accuracy compared to the state-of-the-art methods. Our results largely improve the accuracy by 14.71% and 8.70% and 11.07% compared to the baseline on CUB200 and MIT-67 and Stanford40 respectively, showing that our method can also improve the performance of classification networks on fine-grained classification tasks. Tables 2 and 3 show that our method significantly improves the generalization ability of target networks on various classification tasks.

Scalability. Table 4 shows that our gated ensemble self-teacher structure has good scalability with regard to the increment of ResNet layers. We observe that the training time increases sublinearly with the number of layers of ResNet, which demonstrates good scalability. It is

[1] https://www.kaggle.com/competitions/challenges-in-representation-learning-facial-expression-recognition-challenge/data

Table 2
Performance comparison on CIFAR-100 and TinyImageNet in terms of top-1 classification accuracy. The experiment is repeated three times and reports the mean and standard deviation of accuracy. The best results are in bold. The runner-up results are in underlined.

Dataset	Network	Baseline	ONE	DDGSD	BYOT	CS-KD	FRSKD	Our
CIFAR100	WRN-16-2	71.05 ± 0.42	73.01 ± 0.23	71.96 ± 0.05	71.03 ± 0.15	72.14 ± 0.47	74.10 ± 0.16	**75.59 ± 0.25**
	ResNet18	75.32 ± 0.00	76.66 ± 0.00	76.61 ± 0.47	77.54 ± 0.07	78.22 ± 0.08	77.64 ± 0.34	**79.17 ± 0.11**
TinyImageNet	WRN-16-2	51.71 ± 0.00	52.10 ± 0.20	51.07 ± 0.24	51.26 ± 0.35	50.34 ± 0.12	53.41 ± 0.26	**53.68 ± 0.19**
	ResNet18	55.89 ± 0.00	58.17 ± 0.00	56.46 ± 0.24	58.00 ± 0.20	58.71 ± 0.10	59.31 ± 0.19	**60.99 ± 0.64**

Table 3
Performance comparison on FGVR tasks. The experiment is repeated three times and reports the mean and standard deviation of the accuracy. The best results are in bold. The runner-up results are in underlined.

Dataset	Network	Baseline	ONE	DDGSD	BYOT	CS-KD	FRSKD	Our
CUB200	ResNet18	54.85 ± 0.00	58.93 ± 0.32	58.49 ± 0.55	58.60 ± 0.55	64.34 ± 0.08	65.39 ± 0.13	**69.56 ± 0.20**
MIT-67	ResNet18	55.45 ± 0.00	57.38 ± 0.68	59.00 ± 0.77	54.63 ± 0.45	57.36 ± 0.37	61.74 ± 0.67	**64.15 ± 0.69**
Stanford40	ResNet18	45.77 ± 0.00	45.64 ± 0.71	45.81 ± 1.79	46.29 ± 0.41	47.23 ± 0.22	56.00 ± 1.19	**56.84 ± 0.60**

Table 4
The scalability of gated ensemble of self-teacher networks. Our networks is tested on CIFAR-100 in terms of top-1 classification accuracy. Parameter in table indicates the amount of model parameters, and time is the training time per epoch.

Model	Method	Accuracy	Parameter	Time
ResNet18	baseline	75.32 ± 0.00	10.70MB	60 s
	our	79.17 ± 0.11	13.70MB	90 s
ResNet32	baseline	71.89 ± 0.00	20.36MB	70 s
	our	80.63 ± 0.55	23.36MB	106 s
ResNet50	baseline	73.84 ± 0.03	22.65MB	78 s
	our	81.71 ± 0.13	27.01MB	139 s

clearly that our method yields competitive performance on common public datasets for classification tasks and FGVR tasks, and also has good scalability with less increment of GPU memory consumption. The amount of parameters in our network is only 28% more than the baseline methods.

Robustness to noisy labels. Labels in the real world are often noisy, especially if data is crowd-sourced. To evaluate the robustness against noisy labels, we conduct a series of tests by simulating label noise using randomly assigned labels. From Tables 5 and 6, we can see that our method exceeds other recent advanced methods under noisy labels (as shown in the seventh column). With the increase of noise level (the percentage of samples with noisy labels in the training dataset), the accuracy of all the compared methods decreases significantly while our method is superior to others.

The classification accuracy of the baseline ResNet18 is 75.32% on clean test data of the CIFAR-100 dataset, while it decreases to 67.28%, 60.59%, 53.83%, 42.90% and 29.07% when the noise level increases to 10%, 20%, 30%, 50% and 70% respectively. The proposed gated ensemble self-teacher network achieves the classification accuracy of 79.17% on clean test data of the CIFAR-100 dataset and 73.77%, 66.77%, 59.27%, 52.04% and 40.03% with the noise level to 10%, 20%, 30%, 50% and 70% respectively. Similar observations can be found on the fine-grained MIT-67 dataset with the baseline of ResNet18. These results demonstrate that our gated ensemble self-teacher network has a great robustness to noisy labels.

4.3. Evaluation results of influences-based label noise removal

Evaluation on Noisy Labels. In this section, we study the effect of noisy labels on classification accuracy. Noisy labels degrade the classification accuracy of the network significantly. To improve the classification accuracy, we remove data with noisy labels identified by influences estimation. To validate the effectiveness, we conduct an experiment to flip true labels into other noise ones of a random proportion of training data and then recover the performance by our method. We tried two ways to remove label noise using influences according to Algorithm 1: $\delta < 0$ directly or using k which equals to the number of noisy samples. It is noteworthy that both of them behaves well and recovers the accuracy significantly (as shown in Tables 5 and 6).

The experiments highlight that influences-based label noise removal can effectively eliminate noisy labels. From Table 5, we can see that the CIFAR-100 dataset with noisy labels is raised by 9.25%, 13.54%, 17.85%, 29.19% and 37.38% under 10%, 20%, 30%, 50% and 70% noise level compared to the baseline network, respectively. Similarly in Table 6, it is obvious that the MIT-67 dataset with noisy labels is raised by 14.18%, 19.23%, 23.09%, 29.25% and 26.41% under 10%, 20%, 30%, 50% and 70% noise level compared to the baseline network, respectively. It is appealing that the higher level of noisy labels is, the more performance recovers. Our network consistently keep the accuracy recovered more than 9.25% with the increase of noise level. The time takes to remove the noisy labels using influences estimation depends on the training process. For instance, integrating the influence function into the training on the CIFAR-100 takes 101 s to remove noisy instance. However, it is unnecessary to perform instance removal in every training epoch (every 20 epochs in our experiments), refer to Algorithm 1. In total, the influence function only increases the training time by 3% compared to the training without using the influence function. Similarly, it takes 122 s to the influence function on MIT-67. And the overall training time increases by only 9% compared to not use the influence function at all.

To validate the effectiveness of the proposed influences-based label noise removal, we also conducted experiments on asymmetric noise and achieved decent results as well. As shown in Table 7, our method raise 9.33%, 34.93% and 50.79% under 30%, 50%, and 70% noise level compared to the baseline network on CIFAR-100. Similar performance are obtained on MIT-67. We also conduct an experiment on facial expression recognition to show that our method can be practically applied to defenses against poisoning attacks with mislabeling. The FERC dataset consists of 48 × 48 pixel grayscale images of faces and has seven categories (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral). As shown in Table 8, our influences-based removal can effectively suppress the effect of mislabeling and recover the performance significantly. We demonstrate that our method can be applied as a pluggable defense tool against poisoning attacks with mislabeling.

Effects in Features Space. To verify the impact of the gated ensemble self-teacher network and influences estimation on the classification accuracy, we use t-SNE (Van der Maaten and Hinton, 2008) to analyze the feature embedding of logits of ResNet18 trained on the CIFAR-100 dataset. As shown in Fig. 2, our method is able to separate different classes of data further compared to the baseline, which reveals that our

Table 5
Top-1 classification accuracy under varying levels of noisy labels on the CIRAR-100 dataset. The best results are in bold. The runner-up results are in underlined. Our/− means that influences-based label noise removal was not used. Our/+ means that influences-based label noise removal was used. In the last column, values with asterisks show accuracies of removing noisy instances which have negative influences, i.e., $\delta < 0$, and others are accuracies of removing noisy instances with k (equal to the number of noisy samples simulated). Symmetric label noise: real labels are flipped to other labels with equal probability.

Noise level	Baseline	ONE	BYOT	CS-KD	FRSKD	Our/−	Our/+
0%	75.32 ± 0.00	76.66 ± 0.00	77.54 ± 0.07	78.22 ± 0.08	77.64 ± 0.34	**79.17 ± 0.11**	–
Symmetric-10%	67.28 ± 0.01	71.97 ± 0.27	69.86 ± 0.18	68.94 ± 0.10	72.00 ± 0.13	73.77 ± 0.54	**76.53 ± 0.57** 76.11 ± 0.12*
Symmetric-20%	60.59 ± 0.11	66.65 ± 0.57	63.93 ± 0.28	61.55 ± 0.37	65.74 ± 0.14	66.77 ± 0.59	**74.13 ± 0.10** 72.95 ± 0.57*
Symmetric-30%	53.83 ± 0.62	61.48 ± 0.35	57.14 ± 0.22	54.07 ± 0.37	58.43 ± 0.36	59.27 ± 0.07	**71.68 ± 0.61** 70.59 ± 0.51*
Symmetric-50%	42.90 ± 0.11	44.49 ± 0.29	46.90 ± 0.00	54.07 ± 0.37	51.80 ± 0.81	52.04 ± 0.00	– 72.09 ± 0.51*
Symmetric-70%	29.07 ± 0.70	29.87 ± 0.10	31.20 ± 0.84	41.83 ± 0.60	41.46 ± 0.09	40.03 ± 0.84	– 66.45 ± 0.32*

Table 6
Top-1 classification accuracy under varying levels of noisy labels on the fine-grained MIT-67 dataset. The best results are in bold. The runner-up results are in underlined. Our/− means that influences-based label noise removal was not used. Our/+ means that influences-based label noise removal was used. In the last column, values with asterisks show accuracies of removing noisy instances which have negative influences, i.e., $\delta < 0$, and others are accuracies of removing noisy instances with k (equal to the number of noisy samples simulated). Symmetric label noise: real labels are flipped to other labels with equal probability.

Noise level	Baseline	ONE	BYOT	CS-KD	FRSKD	Our/−	Our/+
0%	55.45 ± 0.00	57.38 ± 0.68	54.63 ± 0.45	57.36 ± 0.37	61.74 ± 0.67	**64.15 ± 0.69**	–
Symmetric-10%	47.71 ± 0.08	50.02 ± 0.32	46.30 ± 0.03	46.51 ± 1.19	57.11 ± 1.72	57.16 ± 1.83	61.89 ± 1.04 **62.42 ± 1.20***
Symmetric-20%	39.50 ± 0.57	43.54 ± 0.52	39.36 ± 0.60	39.87 ± 0.47	49.03 ± 1.42	51.19 ± 0.56	58.73 ± 1.44 **59.51 ± 0.37***
Symmetric-30%	33.35 ± 0.49	36.21 ± 0.99	32.20 ± 0.00	32.93 ± 1.04	43.11 ± 1.24	45.32 ± 0.88	56.44 ± 1.16 **57.05 ± 0.26***
Symmetric-50%	21.42 ± 0.42	22.97 ± 0.14	19.05 ± 0.07	21.82 ± 0.021	28.13 ± 0.04	32.39 ± 0.84	– **50.67 ± 0.18***
Symmetric-70%	12.80 ± 0.15	11.75 ± 0.11	8.9 ± 0.14	11.71 ± 0.16	13.13 ± 0.19	15.97 ± 0.42	– **39.21 ± 0.16***

Table 7
Top-1 classification accuracy under different levels of asymmetric noise labels on CIFAR-100 and fine-grained MIT-67 data sets. The best results are in bold. Asymmetric label noise: real labels are flipped to other labels based on cyclic shifting.

Noise level	0%	Asymmetric-30%	Asymmetric-50%	Asymmetric-70%
CIFAR100	79.17 ± 0.11	61.63 ± 0.80	37.98 ± 0.21	18.16 ± 0.12
		70.96 ± 0.41	**72.91 ± 0.55**	**68.95 ± 0.18**
MIT-67	64.15 ± 0.69	45.93 ± 0.10	29.66 ± 0.79	13.69 ± 0.16
		57.91 ± 0.00	**51.72 ± 0.17**	**40.04 ± 0.10**

Table 8
Top-1 classification accuracy under different levels of asymmetric noise labels on FERC (Facial Expression Recognition Challenge). The best results are in bold.

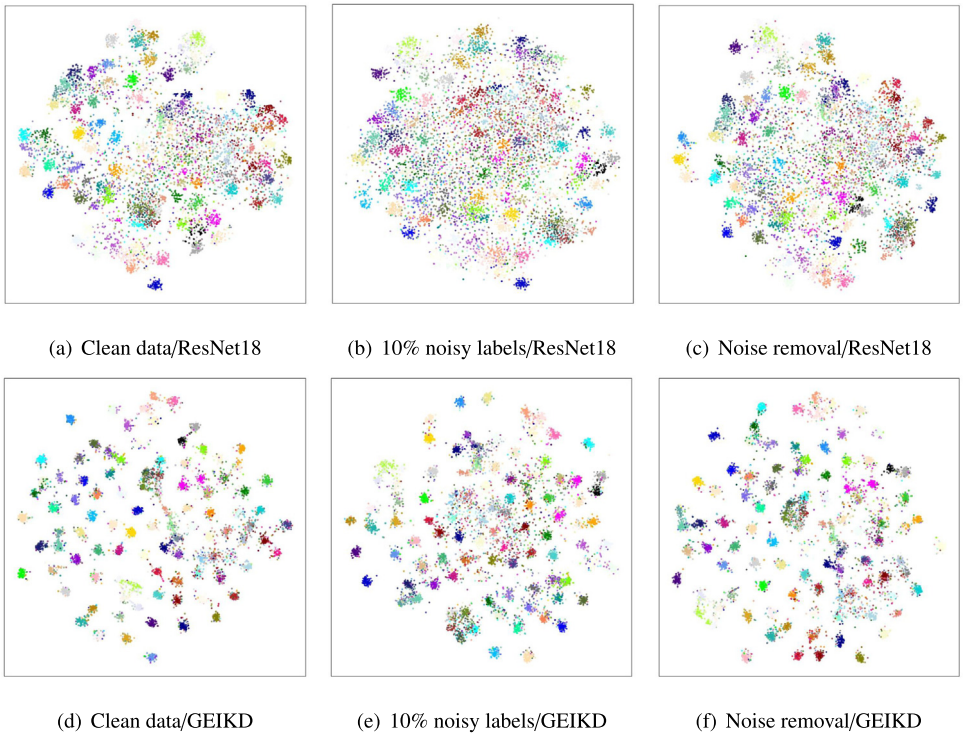
Noise level	0%	30%	50%
FERC	61.23 ± 0.05	51.62 ± 0.09	44.33 ± 0.14
		57.26 ± 0.70	**54.69 ± 0.01**

gated ensemble network gives more refined knowledge which reduces the intra-class distance and behaves more discriminative. As shown in Fig. 2(b), it is clear that data points overlap in many regions without clear classification boundaries because of noisy labels. Ten percentage of samples with noisy labels in training dataset degrade the baseline network significantly, while our gated ensemble self-teacher network still preserves decent inter-class distances for most of clusters as shown in Fig. 2(e). After noise removal by our method, it is obvious that most of clusters are more separable as shown in Fig. 2(f). By comparing with Figs. 2(e) and 2(f), it is reasonable to conclude that our influences-based label noise removal certainly benefits the generalization ability of the classifier. Fig. 2(c) shows that the logits are also improved after using our label noise removal to the baseline network, which demonstrates our plug-and-play property is general for existing networks.

Noisy labels may lead to overfitting the training set, and eventually show poor generalization ability to unseen instances. We can alleviate overfitting by removing label noise based on influences estimation.

4.4. Ablation study

To study the effects of influences-based label noise removal strategy, we provide ablation studies in terms of classification accuracy. In addition to the baseline, we select two advanced methods to verify the effect of influences estimation by plugging them into these networks. We select CIFAR-100 dataset and MIT-67 dataset and use resnet18 as the baseline network.



(a) Clean data/ResNet18 (b) 10% noisy labels/ResNet18 (c) Noise removal/ResNet18

(d) Clean data/GEIKD (e) 10% noisy labels/GEIKD (f) Noise removal/GEIKD

Fig. 2. Visualization of logits of ResNet-18 using t-SNE on CIFAR-100 from (a) to (c) is the visualization method of the baseline method, and from (d) to (f) is the classifier network trained under our proposed self-distillation framework. Where (a) and (d) are results on the clean training set, (b), (e) are results on the training set with 10% noisy labels, and (c), (f) are results on the noisy training set after 10% noisy data removal by our method.

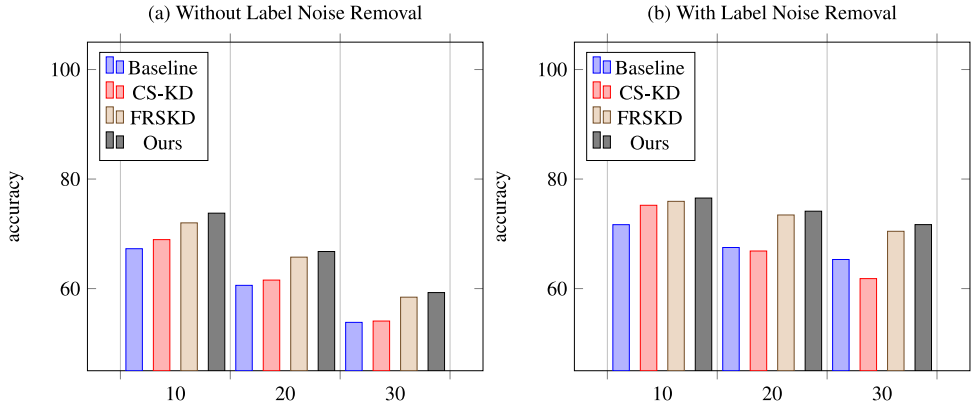


Fig. 3. The effects with and without influence-based removal on CIFAR-100 dataset.

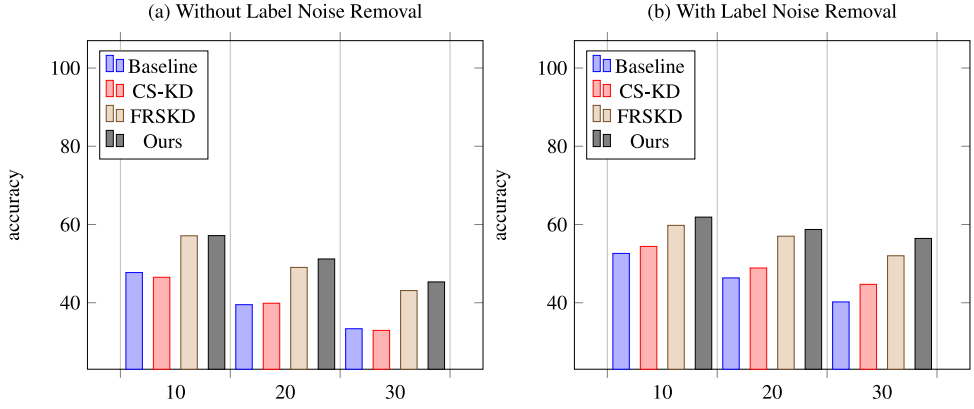


Fig. 4. The effects with and without influence-based removal on MIT-67 dataset.

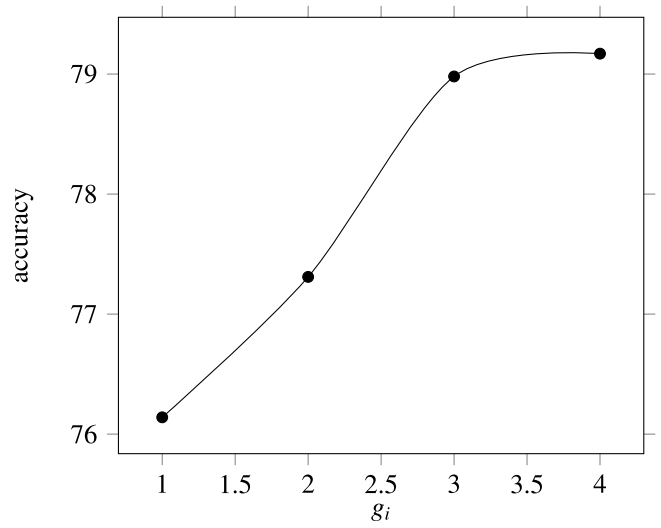


Fig. 5. Ablation study on the number of branches on CIFAR-100 in terms of top-1 classification accuracy. The experiment is repeated three times and reports the mean accuracy.

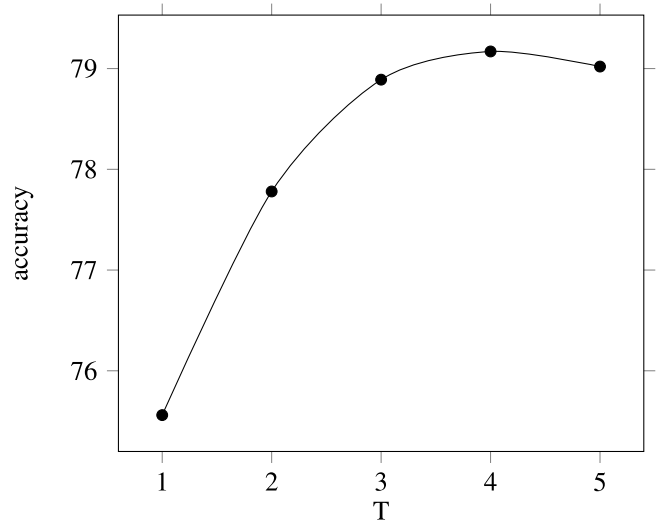


Fig. 6. Ablation study on the temperature parameter on CIFAR-100 in terms of top-1 classification accuracy. The experiment is repeated three times and reports the mean accuracy.

As shown in Figs. 3(a), 4(a), all the four methods have degraded in some degree on both datasets due to the influences of noisy labels. Apparently, our method is always superior to others. Subsequently, we apply the proposed influences-based label noise removal to reject the noisy data on the baseline, *i.e.*, CS-KD and FRSKD networks. We observe that all the methods have been improved significantly, as shown in Fig. 3(b), 4(b). The results fully demonstrate that the proposed influences-based label noise removal can be used as a general module for any existing knowledge distillation methods with various tasks.

We study the effects of branches: from 1 to 4 branches. The ablation study results are presented in Fig. 5. We can observe: (1) multi-branches achieve higher AP scores than the single branch. (2) the AP growth tends to saturate when the number of branches approaches 4.

In our proposed architecture of KD network, two hyperparameters may affect the classification accuracy: the temperature parameter T and λ. As shown in Fig. 6, the performance begins to decline once T exceeds 4. As depicted in Fig. 7, we empirically choose the value of lambda respectively on different datasets.

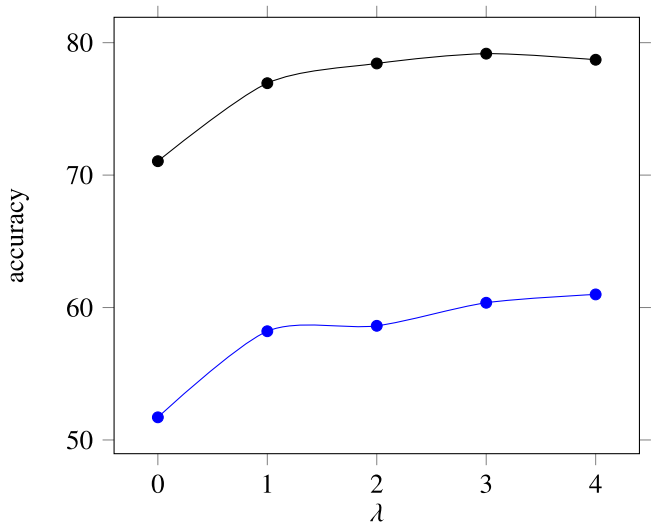


Fig. 7. Ablation study on the hyperparameter λ on CIFAR-100 and TinyImageNet in terms of top-1 classification accuracy. The experiment is repeated three times and reports the mean accuracy. The black line indicates the results on CIFAR-100, the blue line indicates the results on TinyImageNet. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5. Conclusion

Most of current knowledge distillation works have focused on improving knowledge, network architectures and training schemes, and have given insufficient attention to the effect of noise in the training set. To the best of our knowledge, there are few methods discussing the issue of handling noisy labels from poisoning attacks in the field of knowledge distillation. In this paper, we study the vulnerability of the state-of-the-art KD methods against noisy labels. To understand and mitigate the impact of label noise, we propose an effective solution consisting of a gated self-distillation network and influences-based label noise removal. The proposed self-distillation network improves knowledge transferred to the student network with distinct kinds of knowledge from multi-resolution feature maps. In contrast to label noise filtering methods in semi-supervised learning, our distillation training method confronts noisy labels directly and eliminates the samples that play negative influences on the distillation loss as samples with noisy labels. In a word, our approach mitigates the impact of noisy labels on KD networks from both sides, network architectures and distillation schemes. It is worth noting that our influences-based label noise removal provides a standard pluggable tool to any existing KD network to combat against noise or even poison samples. We also demonstrate that our method can be practically applied to defenses against poisoning attacks with mislabeling through experiments on FERC dataset. However, No Free Lunch Theorem points out every optimization has a price. The proposed influences-based noise removal makes extra training efforts on influences estimation. We will make further improvements on computation simplification, such as an optimization for task-specific networks using partial gradients from the critical layers.

CRediT authorship contribution statement

Fuchang Liu: Conceptualization, Writing – original draft, Response to Reviews. **Yu Wang:** Methodology, Software, Validation, Response to Reviews. **Zheng Li:** Data curation, Investigation. **Zhigeng Pan:** Supervision, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant (No. 62072150), the Zhejiang Provincial Natural Science Foundation of China under Grant (No. LZ23F020002).

References

Baek, K., Lee, S., Shim, H., 2022. Learning from better supervision: Self-distillation for learning with noisy labels. In: 2022 26th International Conference on Pattern Recognition. ICPR, IEEE, pp. 1829–1835.

Chen, D., Mei, J.-P., Wang, C., Feng, Y., Chen, C., 2020. Online knowledge distillation with diverse peers. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 3430–3437.

Chung, I., Park, S., Kim, J., Kwak, N., 2020. Feature-map-level online adversarial knowledge distillation. In: International Conference on Machine Learning. PMLR, pp. 2006–2015.

Cordeiro, F.R., Sachdeva, R., Belagiannis, V., Reid, I., Carneiro, G., 2023. Longremix: Robust learning with high confidence samples in a noisy label environment. Pattern Recognit. 133, 109013.

Courbariaux, M., Bengio, Y., David, J.-P., 2015. Binaryconnect: Training deep neural networks with binary weights during propagations. In: Advances in Neural Information Processing Systems.

Denil, M., Shakibi, B., Dinh, L., Ranzato, M., De Freitas, N., 2013. Predicting parameters in deep learning. In: Advances in Neural Information Processing Systems pp. 2148–2156.

Goldberger, J., Ben-Reuven, E., 2017. Training deep neural-networks using a noise adaptation layer. In: International Conference on Learning Representations.

Han, J., Luo, P., Wang, X., 2019. Deep self-learning from noisy labels. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5138–5147.

Han, S., Mao, H., Dally, W.J., 2015a. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149.

Han, S., Pool, J., Tran, J., Dally, W., 2015b. Learning both weights and connections for efficient neural network. In: Advances in Neural Information Processing Systems.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., Sugiyama, M., 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In: Advances in Neural Information Processing Systems.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778.

Heo, B., Kim, J., Yun, S., Park, H., Kwak, N., Choi, J.Y., 2019. A comprehensive overhaul of feature distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1921–1930.

Hinton, G., Vinyals, O., Dean, J., et al., 2015. Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenets: Efficient convolutional neural networks for mobile vision applications. In: In Computer Vision and Pattern Recognition. CVPR.

Iandola, F.N., Han, S., Moskewicz, M.W., Ashraf, K., Dally, W.J., Keutzer, K., 2016. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. In: In International Conference on Learning Representations.

Ji, M., Shin, S., Hwang, S., Park, G., Moon, I.-C., 2021. Refine myself by teaching myself: Feature refinement via self-knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition pp. 10664–10673.

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

Kim, K., Ji, B., Yoon, D., Hwang, S., 2021. Self-knowledge distillation with progressive refinement of targets. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6567–6576.

Koh, P.W., Liang, P., 2017. Understanding black-box predictions via influence functions. In: International Conference on Machine Learning. PMLR, pp. 1885–1894.

Krizhevsky, A., Hinton, G., et al., 2009. Learning multiple layers of features from tiny images. Toronto, ON, Canada.

Li, Z., Huang, Y., Chen, D., Luo, T., Cai, N., Pan, Z., 2020a. Online knowledge distillation via multi-branch diversity enhancement. In: Proceedings of the Asian Conference on Computer Vision.

Li, Z., Li, X., Yang, L., Zhao, B., Song, R., Luo, L., Li, J., Yang, J., 2023. Curriculum temperature for knowledge distillation. In: Proceedings of the AAAI Conference on Artificial Intelligence.

Li, J., Socher, R., Hoi, S.C., 2020b. Dividemix: Learning with noisy labels as semi-supervised learning. arXiv preprint arXiv:2002.07394.

Liang, D., Du, Y., Sun, H., Zhang, L., Liu, N., Wei, M., 2021. Nlkd: Using coarse annotations for semantic segmentation based on knowledge distillation. In: ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing. ICASSP, pp. 2335–2339. http://dx.doi.org/10.1109/ICASSP39728.2021.9414355.

Liu, Y., Cao, J., Li, B., Yuan, C., Hu, W., Li, Y., Duan, Y., 2019. Knowledge distillation via instance relationship graph. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7096–7104.

Malach, E., Shalev-Shwartz, S., 2017. Decoupling" when to update" from" how to update". In: Advances in Neural Information Processing Systems.

Natarajan, N., Dhillon, I.S., Ravikumar, P.K., Tewari, A., 2013. Learning with noisy labels. In: Advances in Neural Information Processing Systems.

Nguyen, D.T., Mummadi, C.K., Ngo, T.P.N., Nguyen, T.H.P., Beggel, L., Brox, T., 2019. Self: Learning to filter noisy labels with self-ensembling. arXiv preprint arXiv:1910.01842.

Novikov, A., Podoprikhin, D., Osokin, A., Vetrov, D.P., 2015. Tensorizing neural networks. In: Advances in Neural Information Processing Systems. pp. 442–450.

Park, W., Kim, D., Lu, Y., Cho, M., 2019. Relational knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3967–3976.

Patrini, G., Rozza, A., Krishna Menon, A., Nock, R., Qu, L., 2017. 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.

Pearlmutter, B.A., 1994. Fast exact multiplication by the hessian. Neural Comput. 6 (1), 147–160.

Quattoni, A., Torralba, A., 2009. Recognizing indoor scenes. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp. 413–420.

Rastegari, M., Ordonez, V., Redmon, J., Farhadi, A., 2016. Xnor-net: Imagenet classification using binary convolutional neural networks. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, the Netherlands, October 11–14, 2016, Proceedings, Part IV. Springer, pp. 525–542.

Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., Rabinovich, A., 2014. Training deep neural networks on noisy labels with bootstrapping. arXiv preprint arXiv:1412.6596.

Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y., 2014. Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550.

Tan, M., Pang, R., Le, Q.V., 2020. Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10781–10790.

Tian, Y., Krishnan, D., Isola, P., 2020. Contrastive representation distillation. In: International Conference on Learning Representation.

Van der Maaten, L., Hinton, G., 2008. Visualizing data using t-SNE. J. Mach. Learn. Res. 9 (11), 2579–2605.

Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S., 2011. The Caltech-Ucsd Birds-200–2011 Dataset. California Institute of Technology.

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., Bailey, J., 2019. Symmetric cross entropy for robust learning with noisy labels. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 322–330.

Xu, T.-B., Liu, C.-L., 2019. Data-distortion guided self-distillation for deep neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence pp. 5565–5572.

Yao, B., Jiang, X., Khosla, A., Lin, A.L., Guibas, L., Fei-Fei, L., 2011. Human action recognition by learning bases of action attributes and parts. In: 2011 International Conference on Computer Vision. IEEE, pp. 1331–1338.

Yuan, L., Tay, F.E., Li, G., Wang, T., Feng, J., 2020. Revisiting knowledge distillation via label smoothing regularization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3903–3911.

Yun, S., Park, J., Lee, K., Shin, J., 2020. Regularizing class-wise predictions via self-knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13876–13885.

Zagoruyko, S., Komodakis, N., 2016. Wide residual networks. arXiv preprint arXiv:1605.07146.

Zhang, L., Song, J., Gao, A., Chen, J., Bao, C., Ma, K., 2019. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision pp. 3713–3722.

Zhu, X., Gong, S., et al., 2018. Knowledge distillation by on-the-fly native ensemble. In: Advances in Neural Information Processing Systems.