IEICE TRANS. INF. & SYST., VOL.E107–D, NO.1 JANUARY 2024

LETTER *Special Section on Enriched Multimedia — Media technologies opening up the future —*

Unbiased Pseudo-Labeling for Learning with Noisy Labels

Ryota HIGASHIMOTO[†], ***Nonmember***, **Soh YOSHIDA**[†a)], ***Member***, **Takashi HORIHATA**[†], ***Nonmember***, ***and*** **Mitsuji MUNEYASU**[†], ***Fellow***

SUMMARY Noisy labels in training data can significantly harm the performance of deep neural networks (DNNs). Recent research on learning with noisy labels uses a property of DNNs called the memorization effect to divide the training data into a set of data with reliable labels and a set of data with unreliable labels. Methods introducing semi-supervised learning strategies discard the unreliable labels and assign pseudo-labels generated from the confident predictions of the model. So far, this semi-supervised strategy has yielded the best results in this field. However, we observe that even when models are trained on balanced data, the distribution of the pseudo-labels can still exhibit an imbalance that is driven by data similarity. Additionally, a data bias is seen that originates from the division of the training data using the semi-supervised method. If we address both types of bias that arise from pseudo-labels, we can avoid the decrease in generalization performance caused by biased *noisy* pseudo-labels. We propose a learning method with noisy labels that introduces unbiased pseudo-labeling based on causal inference. The proposed method achieves significant accuracy gains in experiments at high noise rates on the standard benchmarks CIFAR-10 and CIFAR-100.

key words: *deep learning, learning with noisy labels, semi-supervised learning, causal inference*

1. Introduction

The remarkable success of deep neural networks (DNNs) is due to the collection of large datasets with human-annotated labels. However, such reliable labeling is expensive and time-consuming. By contrast, inexpensive alternatives exist for collecting labeled data. These inexpensive methods inevitably collect samples with noisy labels. Recent research [1] has shown that DNNs tend to overfit noisy labels, resulting in poor generalization performance.

Research on learning with noisy labels (LNL) has confirmed that DNNs first learn easy (most likely clean) samples and then learn hard (most likely noisy) samples. This is called the *memorization effect* [2], which has been widely validated. A simple and practical approach to exploiting the memorization effect is sample selection [3], which monitors model losses and selects small-loss samples to avoid the influence of unreliable samples more likely to be noisy labels. Co-teaching [3] trains two models on each other, selecting samples with a small loss in one model to train the other.

The strategy that combines LNL with semi-supervised learning (SSL) is one of the main reasons for the progress of LNL. Training data in SSL consist of a small number of labeled data and many unlabeled data. Pseudo-labels are generated from the confident predictions of the model trained on the labeled samples and assigned to the unlabeled samples. MixMatch [4] takes advantage of unlabeled samples by forcing the model to make consistent predictions on unlabeled samples that have been augmented using different weak data augmentation techniques. FixMatch [5] generates pseudo-labels from weakly augmented unlabeled samples and forces the model to match the output for strongly augmented unlabeled samples to the pseudo-labels. By employing a threshold, this method ensures it uses only reliable pseudo-labels, which significantly improves its performance in SSL. However, concerns have been reported regarding *model bias* in SSL, specifically that pseudo-labels can be forced into imbalance because of data similarity, even when the models have been trained on balanced data [6].

The imbalanced pseudo-labeling problem, which should be related to noisy pseudo-labels, has not been discussed in the LNL field. DivideMix [7] is a pioneering method that combines sample selection and SSL for LNL and has achieved state-of-the-art performance in recent years by dividing the training data into a set of labeled and unlabeled samples and assigning pseudo-labels to the unlabeled samples, which is called co-divide. The development of advanced state-of-the-art methods based on DivideMix is an active research topic [8]. However, DivideMix, when dividing the training data using a model that has memorized noisy labels, can lead to bias in which the number of labels for certain classes either increases or decreases according to the errors in the memorized labels. We refer to this phenomenon as *data bias*. This problem is particularly prominent under conditions with high noise rates. In this letter, we first investigate the generation of imbalanced pseudo-labels even in DivideMix, propose an unbiased pseudo-labeling, and show experimentally that removing model and data bias benefits LNL.

In summary, our contribution is three-fold.

- To the best of our knowledge, this letter is the first to demonstrate that imbalanced pseudo-labeling caused by model and data bias reduces robustness to noisy labels in supervised learning that employs the SSL strategy.
- We propose a dual-model bias estimation method based on causal inference that concurrently tackles both types of bias. This method realizes the concurrent mitigation

Manuscript received March 15, 2023.
Manuscript revised August 4, 2023.
Manuscript publicized September 19, 2023.
[†]The authors are with the Faculty of Engineering Science, Kansai University, Suita-shi, 564–8680 Japan.
a) E-mail: sohy@kansai-u.ac.jp (Corresponding author)
DOI: 10.1587/transinf.2023MUL0002

of bias across two networks. Our identification of data bias, another significant factor in SSL-based LNL methods, enhances our comprehension of how bias manifests in pseudo-labeling.

- Experiments on several benchmarks with different noise types and noise rates show that the proposed method significantly improves the performance of the conventional method, providing insight into improving the performance of LNL.

This study presents an analysis of two distinct types of bias within SSL in the context of LNL: model bias and data bias. Notably, we introduce data bias as an issue yet to be thoroughly explored in SSL-based LNL. Data bias arises from inaccurate memorization of noisy labels, leading to an imbalanced distribution of labels, especially in environments with high levels of noise.

The proposed method in this letter is the dual-model bias estimation method. This strategy employs outputs from two concurrently trained DNNs used in the co-divide process, facilitating the simultaneous management of both model and data bias. The effectiveness of this method extends to environments where the direct application of the method proposed by [6] may be inadequate, specifically in LNL situations characterized by high noise rates.

The novelty of our method stems from the divergent roles of the two networks. In contrast to the unidirectional teacher–student relationship in the method of [6], our approach relies on mutual interdependence between the networks, resulting in a mutual optimization process. This distinct approach affords a more encompassing bias estimation and effective bias reduction. Consequently, our method is not merely an application of the method proposed by [6], but a useful extension.

2. Biased Pseudo-Labeling in LNL

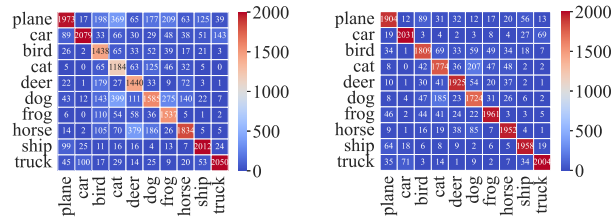
DivideMix. We briefly review DivideMix, which utilizes co-divide, a process that trains two networks simultaneously. First, each network is warmed up for several epochs using the cross-entropy $\mathbf{H}(\cdot)$ with batch size B formulated as follows:

$$\mathcal{L}_s^{(k)} = \frac{1}{B}\sum \mathbf{H}(\mathbf{y}, \mathbf{p}^{(k)}), \tag{1}$$

where $k \in \{1, 2\}$ is the model number, $\mathbf{y}$ and $\mathbf{p}^{(k)}$ denote a one-hot label and the model prediction, respectively. Then, for each network, a Gaussian mixture model is fitted to the loss distribution of each sample, and the training set is divided into labeled (clean) and unlabeled (noisy) data. The separated datasets are used to train the networks in the next epoch. The set of unlabeled samples is assigned pseudo-labels generated from the model's predictions, represented by the following equation:

$$\bar{\mathbf{y}} = \frac{1}{2}(\mathbf{p}^{(1)} + \mathbf{p}^{(2)}), \tag{2}$$

$$\hat{\mathbf{y}} = \bar{\mathbf{y}} \Big/ \sum\nolimits_{c=1}^{C} \bar{y}_c\ , \tag{3}$$



(a) DivideMix (b) Proposed

Fig. 1 Confusion matrices for pseudo-labels (rows: ground truth, columns: classes of pseudo-labels)

where C is the number of classes. The loss for unlabeled samples is formulated as follows:

$$\mathcal{L}_u^{(k)} = \frac{1}{B}\sum ||\hat{\mathbf{y}} - \mathbf{p}^{(k)}||_2^2. \tag{4}$$

The final loss function consists of two terms: $\mathcal{L}^{(k)} = \mathcal{L}_s^{(k)} + \lambda_u \mathcal{L}_u^{(k)}$, where λ_u is a hyperparameter. Note that the derivation of $\mathcal{L}$ is associated with MixMatch.

Bias in DivideMix. DivideMix treats samples with labels likely to be noisy labels as unlabeled samples and generates pseudo-labels from the confident predictions of the trained model. However, especially at high noise rates, the model fits the noisy labels, and the predictions are unreliable and generate incorrect (biased) pseudo-labels. This biased pseudo-labeling occurs when the model learns noisy labels as clean labels and outputs incorrect predictions as confident predictions based on incorrect knowledge. This is referred to as model bias in LNL.

To verify the model bias in LNL, DivideMix and the proposed method were used to obtain pseudo-labels for CIFAR-10 when 90% of the dataset was contaminated by noisy labels. Figure 1 shows the resulting confusion matrices at the point of best performance during training (details are given in Sect. 4.1). Symmetric noise, which randomly flips the labels of training samples to one of the other classes with a certain probability, was used.

Figure 1 (a) shows the confusion matrix for DivideMix, which generates incorrect pseudo-labels such as dog→cat and horse→deer. The visual similarities between these classes cause these errors, and prior research [6] has stated that such pseudo-label bias is due to model bias. However, incorrect pseudo-labels such as plane→cat and dog→frog were not reported in [6]. These errors are caused by data bias that appears due to noisy labels. More specifically, when Model 1 memorizes a noisy label such as plane→cat, the data labeled as "cat" yields a lower loss value due to the prediction–label agreement, while the data labeled as "plane" results in a higher loss value due to the prediction–label discordance. As a result, the clean label dataset identified by the model becomes skewed, containing more "cat" labels and fewer "plane" labels. When this skewed data is used to train Model 2, its predictions also become biased. By contrast, as shown in Fig. 1 (b), the pseudo-labeling errors and bias toward specific classes are reduced by our approach. We present the method for achieving this in the next section.

3. Method

3.1 Preliminaries

Causal inference uses factual assumptions alone to draw counterfactual hypothetical conclusions [9]. Wang et al. [6] consider the undesirable model bias counterfactual and devise counterfactual reasoning [10] that dynamically reduces this effect. In counterfactual reasoning, undesirable model biases become factual assumptions and incorporate causal relationships that generate erroneous predictions. In Fig. 2, when A is a particular sample $A = A_i$, a direct causal relationship along $A_i \rightarrow Y_i$, unaffected by model bias, is defined as a controlled direct effect (CDE) [10] as follows:

$$CDE(Y_i) = [Y_i|do(A_i), do(D)] - [Y_i|do(\hat{A}), do(D)], \quad (5)$$

where the $do(\cdot)$ operator represents a hypothetical scenario in which the variable is fixed to a specific value so that the variable intervenes directly in the prediction without model bias, and $\hat{A} = \{A_1, \ldots A_i, \ldots, A_n\}$ denotes all samples with n samples. When all samples $A = \hat{A}$ are exposed under mediator (model) D with fixed model parameters (denoted as $do(\hat{A})$), the direct causal effect of individual samples is lost from the observed average output and the model bias is regarded as an indirect effect of Y. By contrast, when $A = A_i$ is exposed (denoted as $do(A_i)$), Y includes the indirect effect of model bias, but D retains the direct causal effect. Thus, if we remove the indirect effect of the model $[Y_i|do(\hat{A}), do(D)]$ from the model output $[Y_i|do(A_i), do(D)]$ for $A = A_i$ as in Eq. (5), we obtain a CDE, that is, predictions without the model bias effect.

3.2 Unbiased Pseudo-Labeling for LNL

Section 2 shows that using DivideMix with the SSL strategy produces model and data bias. This bias causes unlabeled samples to be assigned incorrect pseudo-labels again, leading to a vicious cycle of reduced model generalizability. We propose a semi-supervised LNL that introduces unbiased pseudo-labeling using counterfactual inference. Figure 3 shows an overview of the unbiased pseudo-labeling introduced in our proposed LNL. Rather than estimating bias independently in the two models, our method exploits the outputs from both models to estimate model and data bias. By removing the pseudo-label bias, unlabeled samples can be effectively utilized, improving the model's generalization performance and enabling train models to be robust to noisy labels.

Measuring the counterfactual results for all unlabeled samples u_i ($\in \mathcal{U}$; $\mathcal{U}$ is a set of unlabeled samples) is computationally expensive. We follow [6] and approximate the CDE using the approximated CDE (ACDE). Previous methods use a single model for bias estimation and pseudo-labeling, and when incorporated into DivideMix, two models are used to perform independent model bias estimation. However, independent bias estimation fails to debias because it does not consider data bias. Model bias occurs when the model generates false pseudo-labels due to data similarity and memorizes these false pseudo-labels, biasing the pseudo-labels. Data bias, by contrast, occurs when the model memorizes noisy labels, biasing the data such that some classes have more or fewer labels, as discussed in Sect. 2. In addition, biased data can cause the training model to generate biased pseudo-labels. Both data bias and model bias can be manifested by the memorization of noisy labels. The data bias in DivideMix is caused by losses computed from the outputs of different models. That is, the bias of the other model causes the data bias. Therefore, it is possible to estimate the bias in a model from its own output and estimate the data bias from the output of the other model. In our method, we use the outputs of both models for bias estimation. Thus, we estimate model bias and data bias comprehensively as follows:

$$\hat{\mathbf{p}} \leftarrow m\hat{\mathbf{p}} + (1-m)\frac{1}{2B}\sum_{b=1}^{B}\left(\mathbf{p}_b^{(1)} + \mathbf{p}_b^{(2)}\right), \quad (6)$$

where $\hat{\mathbf{p}}$ is a probability vector containing the estimated model bias and data bias, $\mathbf{p}_b^{(k)}$ is the probability distribution of the unlabeled sample $\alpha(u_b)$ with weak data augmentation, $m \in [0, 1)$ is momentum, and B is the mini-batch size of the unlabeled sample. Unbiased pseudo-labeling using ACDE as an alternative to Eq. (5) is formulated as follows:

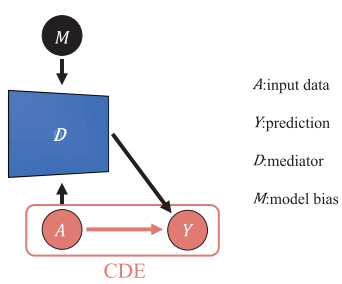


Fig. 2 Causal graph and control direct effect

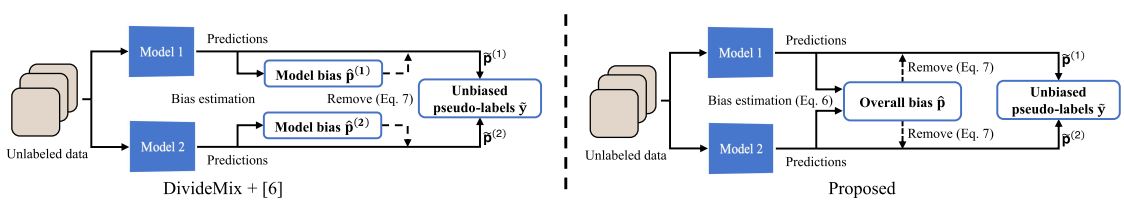


Fig. 3 Overview of unbiased pseudo-labeling

$$\tilde{\mathbf{f}}_i^{(k)} = f_k(\alpha(u_i)) - \lambda \log \hat{\mathbf{p}}, \tag{7}$$

where λ is the hyperparameter that controls the strength of bias removal and $f_k(\cdot)$ is the logit. Finally, we perform unbiased pseudo-labeling by replacing $\mathbf{p}^{(k)}$ in Eq. (2) with unbiased prediction $\tilde{\mathbf{p}}^{(k)}$, which is obtained by applying the *softmax*$(\cdot)$ operator to $\tilde{\mathbf{f}}_i^{(k)}$. The unbiased pseudo-labels are represented by $\tilde{\mathbf{y}}$.

4. Experiments

4.1 Experimental Setup

Dataset. We conducted our experiments by synthesizing noisy labels on CIFAR-10 and CIFAR-100 [11], the standard benchmark datasets in image classification. CIFAR-10 and CIFAR-100 consist of 50k training data and 10k test data with image size $32 \times 32 \times 3$. Following [3], [7], we used two different noise patterns: symmetric and asymmetric. Symmetric noise is a pattern in which labels are randomly flipped at a specified rate regardless of label content. Asymmetric noise, in contrast, flips the labels of similar classes (CIFAR-10:truck→ automobile, bird→ airplane, etc.). Note that following [7], only symmetric noise was used in CIFAR-100.

Experiment Details. We compared the performance of the proposed method with that of DivideMix. An 18-Layer PreAct ResNet [12] was used as the architecture along with SGD with a momentum of 0.9 and weight decay of 5×10^{-4}. The model was trained for 300 epochs with a batch size of 128. We set the initial learning rate to 0.02 and reduced it by a factor of 10 at 150 epochs. The existing parameters were fixed and equal, and the parameters added by the proposed method were set to $m = 0.997$ and $\lambda = 1$, respectively.

4.2 Results

Table 1 presents the experimental results on CIFAR-10 with symmetric noise at rates of 20%, 50%, 80%, and 90% as well as asymmetric noise at 40%. We use the best test accuracy over all epochs (Best) and the average test accuracy of the last 10 epochs (Last) for comparison. The proposed method outperforms DivideMix at all noise rates. At a 90% noise rate, the proposed method improves DivideMix accuracy by about 14%. Table 2 shows the experimental results for CIFAR-100 with symmetric noise at rates of 20%, 50%, 80%, and 90%. On CIFAR-100, the proposed method outperforms DivideMix at all noise rates. These experimental results suggest that the proposed method is particularly effective in situations with high noise rates and few correct labels. On CIFAR-10, the proposed method achieves almost the same test accuracy values even when the noise rate increases. By contrast, on CIFAR-100, the test accuracy drops as the noise rate increases. This problem is thought to be due to the increased influence of model bias caused by a higher number of classes. Furthermore, on CIFAR-100, the batch size of unlabeled data is smaller than the number of classes, and it is possible that the effect of model bias is not sufficiently estimated. Therefore, adaptive control of batch size according to the number of classes is a future issue to be addressed.

Figure 4 shows the accuracy of the pseudo-labels (top row) and the area under the curve (AUC) of the clean/noisy label classification at the SSL stage (bottom row) at each epoch when the symmetric noise rate is varied on CIFAR-10. The AUC indicates whether the labels were divided correctly and whether the model did not divide them incorrectly with high confidence. The proposed method improves the pseudo-label accuracy the most at a noise rate of 90%, significantly increasing the AUC. The proposed method also slightly improves pseudo-label accuracy at noise rates of 50% and 80%, improving the test accuracy, and the AUC is increased accordingly. The improvement in pseudo-label accuracy at a 50% noise rate is slightly higher than that at a 80% noise rate. This difference also affects the AUC results; the improved range of the AUC at a noise rate of 50% is larger than the improved range of the AUC at a noise rate of 80%. As a result, the test accuracy is better at a 50% noise rate, but the accuracy is substantially better than these results at a 90% noise rate. These results suggest that imbalanced pseudo-labels affected by model bias reduce the segmentation accuracy of the training data and the generalization to noisy labels.

4.3 Ablation Study

We conducted an ablation study on bias estimation using CIFAR-100. We proposed a dual-model bias estimation method for both model and data bias, which differs from the previous approach of independent bias estimation. Thus, we contrast the effects of solely removing data bias, only removing model bias, and removing both simultaneously.

When we only remove data bias, we subtract the bias estimated from the secondary model's output. Conversely, when only model bias is targeted, the bias calculated from our primary model's output is removed. Finally, when both model and data bias are addressed, we remove the calculated bias from each model's output.

Table 1 Test accuracy on CIFAR-10

Noise type		Sym.				Asym.
Noise ratio		20%	50%	80%	90%	40%
DivideMix	Best	**96.2**	94.7	93.8	78.0	93.4
	Last	**96.0**	94.5	93.3	76.7	92.5
Proposed	Best	**96.2**	**95.8**	**94.2**	**92.0**	**93.6**
	Last	**96.0**	**95.6**	**94.0**	**91.7**	**92.8**

Table 2 Test accuracy on CIFAR-100

Noise ratio		20%	50%	80%	90%
DivideMix	Best	77.4	74.6	59.7	31.7
	Last	76.9	74.4	59.3	30.8
Proposed	Best	**77.7**	**76.1**	**61.0**	**32.0**
	Last	**77.3**	**75.8**	**60.7**	**31.7**

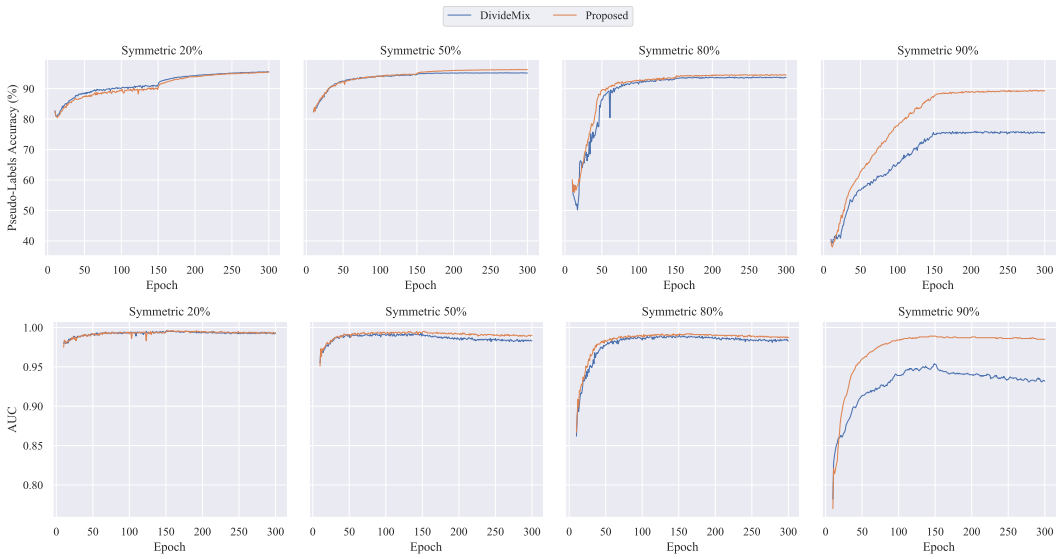


Fig. 4 Accuracy of pseudo-labels and AUC of clean/noisy labels classification on CIFAR-10.

Table 3 Ablation study results for bias estimation on CIFAR-100

Bias type			Noise ratio			
Data bias	Model bias		20%	50%	80%	90%
✗	✗	Best	77.4	74.6	59.7	31.7
		Last	76.9	74.4	59.3	30.8
✓	✗	Best	**77.8**	75.7	**61.2**	31.8
		Last	**77.4**	75.3	**61.1**	31.5
✗	✓	Best	77.6	76.0	60.8	31.8
		Last	77.2	75.7	60.4	31.5
✓	✓	Best	77.7	**76.1**	61.0	**32.0**
		Last	77.3	**75.8**	60.7	**31.7**

The results for the scenario in which no bias is removed aligns closely with the results from DivideMix. However, the result of removing model bias alone mirrors the outcome of integrating [6] with DivideMix. As shown in Table 3, eliminating both model and data bias consistently enhances accuracy across all noise rates. While addressing data bias alone shows promising results, tackling both model and data bias simultaneously offers optimal outcomes in certain cases.

5. Conclusion

In this letter, we indicated that imbalanced pseudo-labels caused by model bias affect the training of robust models on noisy labels. We further proposed a method to remove model bias from imbalanced pseudo-labels to improve the accuracy of the pseudo-labels and the accuracy of training data segmentation. Experimental results show that the proposed method improves the performance of SSL-based LNLs. This letter's insights immediately apply to recent SSL-based LNL methods and will subsequently contribute to their improvement.

Acknowledgments

This work was supported by JSPS KAKENHI (19K01044 and 22K18007).

References

[1] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning requires rethinking generalization," Proc. Int. Conf. Learn. Represent., pp.1–15, 2017.

[2] D. Arpit, S. Jastrzundefinedbski, N. Ballas, D. Krueger, E. Bengio, M.S. Kanwal, T. Maharaj, A. Fischer, A. Courville, Y. Bengio, and S. Lacoste-Julien, "A closer look at memorization in deep networks," Proc. Int. Conf. Machin. Learn., pp.233–242, 2017.

[3] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," Advan. Neu. Inf. Proc. Syst., pp.8536–8546, 2018.

[4] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C.A. Raffel, "MixMatch: A holistic approach to semi-supervised learning," Advan. Neu. Inf. Proc. Syst., pp.5049–5059, 2019.

[5] K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C.A. Raffel, E.D. Cubuk, A. Kurakin, and C.L. Li, "FixMatch: Simplifying semi-supervised learning with consistency and confidence," Advan. Neu. Inf. Proc. Syst., pp.596–608, 2020.

[6] X. Wang, Z. Wu, L. Lian, and S.X. Yu, "Debiased learning from naturally imbalanced pseudo-labels," Proc. IEEE Conf. Comput. Vis. Pattern Recog., pp.14647–14657, 2022.

[7] J. Li, R. Socher, and S.C. Hoi, "DivideMix: Learning with noisy labels as semi-supervised learning," Proc. Int. Conf. Learn. Represent., 2020.

[8] Y. Nomura and T. Kurita, "Consistency regularization on clean samples for learning with noisy labels," IEICE Trans. Inf. & Syst., vol.E105-D, no.2, pp.387–395, 2022.

[9] S. Greenland, J. Pearl, and J.M. Robins, "Confounding and collapsibility in causal inference," Stat. Sci., vol.14, no.1, pp.29–46, 1999.

[10] J. Pearl, "Direct and indirect effects," arXiv preprint arXiv:1301.2300, 2013.

[11] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Master's thesis, Department of Computer Science, University of Toronto, 2009.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," Euro. Conf. Comput. Vis., vol.9908, pp.630–645, 2016.