ITM Web of Conferences **73**, 02035 (2025) https://doi.org/10.1051/itmconf/20257302035
IWADI 2024

Error Correction for Semi-Supervised Classification Based on Fix Match

Ziqian Zhang*

The College of Engineering, Computing and Cybernetics, The Australian National University (ANU), Canberra, ACT 2601, Australia

Abstract. Semi-supervised learning (SSL) leverages unlabeled data to support model training, thereby improving model accuracy. However, most existing SSL methods rely heavily on unlabeled data for model correction while often overlooking the potential error correction capabilities of labeled data. In this paper, we propose Correction FixMatch, which harnesses the error correction potential of labeled data to achieve higher accuracy on the test set. Correction FixMatch is based on the FixMatch model but introduces a novel error-correction mechanism utilizing labeled data. In particular, a training set and a correction set are separated from the labeled dataset. The model is initially trained using the training set, and after a predefined number of steps, the correction set is employed to refine the model through error correction. According to experimental results, the suggested approach performs noticeably better in terms of accuracy than the original FixMatch model under identical testing conditions. This method opens up new possibilities for improving semi-supervised learning models.

1 Introduction

The rapid advancement of deep learning has demonstrated remarkable performance across numerous supervised learning tasks, particularly in image classification [1]. However, methods relying on labeled data face a significant challenge: the cost of data annotation. High-quality annotations not only require domain expertise and substantial time investment but are also susceptible to human error. As a result, enhancing models to effectively utilize a limited amount of labeled data alongside a substantial quantity of unlabeled data has emerged as a significant focus in deep learning research.

Semi-supervised learning requires significantly less labeled data compared to fully supervised learning, while achieving higher accuracy than unsupervised learning by leveraging a small amount of labeled data. Semi-supervised learning (SSL) alleviates the cost of data annotation. As a result, it has been widely applied in popular areas of artificial intelligence, such as natural language processing (NLP), object detection, object classification, semantic segmentation, and others [1-4]. Therefore, effectively utilizing unlabeled samples to improve model performance has become a central challenge in semi-supervised learning.

* Corresponding author: u7545724@anu.edu.au

Many deep learning models have been used in SSL problems because of deep learning's exceptional performance. Tarvainen et al. proposed the Mean Teacher model, which utilizes a teacher-student framework and achieved an accuracy of 90.81% on the CIFAR-10 benchmark dataset using only 4,000 labeled samples in a semi-supervised learning setting [5]. Berthelot et al. proposed the MixMatch model, which integrates consistency regularization, pseudo-labeling, entropy regularization, and MixUp techniques, representing a comprehensive advancement in semi-supervised learning. MixMatch was able to increase test accuracy to 93.58% on the same dataset with the same quantity of labeled data.[3]. Subsequently, Sohn et al. further improved the test accuracy to 95.74% with the FixMatch model, demonstrating its superior performance and reliability [6].

However, most semi-supervised learning models focus on utilizing unlabeled data for model refinement. A common strategy among these methods involves generating pseudo-labels from unlabeled samples and leveraging the model's predictions as pseudo-labels to improve the accuracy of predictions on unlabeled data [5-7]. Such methods often focus on extracting information from unlabeled data to improve the model's learning capabilities, while placing less emphasis on the role of labeled data in error correction. This work, inspired by FixMatch, proposes an improved version that effectively incorporates labeled data for model error correction.

2 Related work

The majority of semi-supervised image classification algorithms now rely on pseudo-labeling and consistency regularization techniques to take advantage of unlabeled data and improve overall classification performance because of their exceptional performance in semi-supervised tasks in recent years.

Optimizing models with a small amount of labeled data and a large volume of unlabeled data has become a key research focus in deep learning Predicting approximation labels for unlabeled data is the fundamental concept of pseudo-labeling. In simple terms, after the model has been trained on labeled data to a certain extent, it predicts the unlabeled data and selects the class with the highest probability as its pseudo-label [8]. Cross-entropy loss is then used to update the model parameters. Pseudo-labeling greatly expands the training data set and aids the model in gradually improving its predictions by adding the anticipated outcomes of unlabeled data as "true labels" to the training set.

Consistency regularization refers to the idea that a model's predictions should remain consistent when faced with different input transformations of the same data, such as data augmentation. In other words, even when an image transforms flipping, cropping, or brightness adjustment, the predicted class should remain unchanged. This technique was first introduced by Bachman et al. and has been widely adopted and further developed in subsequent semi-supervised learning models [5-7, 9]. By applying consistency regularization, we can apply different perturbations to unlabeled data and require the model to make consistent predictions despite these input variations, thereby turning the unlabeled data into effective training signals.

In later research, consistency regularization became widely adopted by many semi-supervised learning models and has been increasingly combined with other techniques [5-7, 10]. Pseudo-labeling and consistency regularization were combined to create hybrid techniques like FixMatch and MixMatch [5, 6]. FixMatch is an example of a hybrid approach that uses consistency regularization with pseudo-labeling to enhance semi-supervised learning performance in a straightforward yet efficient manner [5]. Both strong and weak augmentations are used by FixMatch to provide predictions for unlabeled data. From the weakly augmented data, high-confidence predictions are chosen as pseudo-labels. The model's capacity to generalize is then enhanced by making predictions for the strongly augmented data that, per consistency regularization, should continue to be consistent with those of the weakly augmented data.

3 Method

FixMatch significantly simplifies the approach to semi-supervised learning by combining pseudo-label generation and consistency regularization, enabling unlabeled data to effectively participate in model training and substantially reducing dependence on labeled data. However, FixMatch primarily focuses on the role of unlabeled data in correcting model predictions, while underutilizing the error correction potential hidden within labeled data.

To address this limitation, this paper proposes an improved method that leverages the latent error correction capability of labeled data during model training. Specifically, a training set and a correction set are created from the labeled data at first. The model is first trained using the training set, which aids in its acquisition of fundamental characteristics and patterns. During this process, unlabeled data also participates in pseudo-label generation to maintain the semi-supervised nature of the FixMatch framework. After the model has been trained for a certain number of steps, the correction set is periodically introduced into the training process at fixed intervals to refine and optimize the model more accurately. Figure 1 illustrates the general layout of the proposed method.

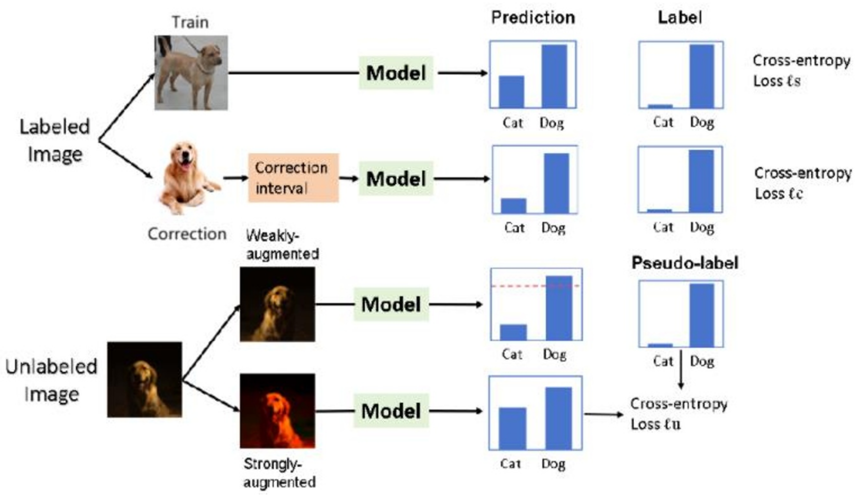

Fig. 1. Model structure (Photo/Picture credit: Original).

The proposed model is largely similar to the FixMatch framework but introduces a key modification: There are two sets of labeled images: one for training and one for correction. The model is first trained using the training set. The correction set is used to improve the model after a predetermined number of training steps. Cross-entropy loss is used in the corrective process to increase the accuracy of the model.

3.1 Dataset

The popular CIFAR-10 benchmark dataset is used in this study to assess the efficacy and performance enhancements of the proposed method. There are 60,000 images total in the CIFAR-10 dataset, which is divided into 10 classes, each of which has 6,000 color images. To better simulate real-world semi-supervised learning scenarios, the experiment employs the same setup as FixMatch, utilizing 4,000 labeled samples—randomly selecting 400 labeled images per class—as the supervised portion for training. The remaining unlabeled data serves as the unsupervised portion, supporting pseudo-label generation and consistency training.

3.2 Loss function

The proposed approach shares similarities with FixMatch in methodology, employing cross-entropy as a key loss component, but introduces notable differences in implementation. The FixMatch loss function is defined as:

$$\mathrm{Loss} = \mathrm{L}_s + \lambda_u * \mathrm{L}_u \tag{1}$$

where L_s represents the supervised loss calculated on labeled data, L_u is the unsupervised loss, and λ_u is a weight factor for the unsupervised loss. In contrast, this study adopts a two-stage training strategy to progressively optimize model performance.

In the early training stage, the model primarily uses the training set and its corresponding supervised loss, employing cross-entropy to help the model learn basic features and patterns. During this stage, the loss function is defined as:

$$\mathrm{Loss}_1 = \mathrm{L}_s + \lambda_\mathrm{u} * \mathrm{L}_u \tag{2}$$

where L_s and L_u retain the same definitions as above.

As training progresses, the correction set is introduced to provide precise calibration and optimization for the model. In this later stage, the loss function is modified to:

$$\mathrm{Loss}_2 = \lambda_\mathrm{u} * \mathrm{L}_\mathrm{u} + \lambda_\mathrm{c} * \mathrm{L}_\mathrm{c} \tag{3}$$

where L_c represents the correction loss calculated using the correction set, and λ_c is the weight factor for the correction loss.

This phased optimization strategy enables the model to focus on learning foundational features in the early stage while progressively improving through error correction in the later stage, thus enhancing its generalization capability and accuracy.

3.3 Experimental setup

FixMatch employs a total of 2^{20} training steps with evaluations conducted every 1,024 steps. However, due to hardware and time constraints, this study reduces the total training steps to 2^{14} (131,072 steps). Evaluations are performed every 128 steps to monitor the model's progress on the validation set, resulting in a total of 128 evaluations. Each training step uses a batch size of 64.

For the correction set, the labeled data was initially weakly augmented and divided into training and correction sets in fixed proportions. The model applied corrections after fixed training intervals. However, this approach yielded suboptimal results because splitting the labeled data reduced the size of the training set, leading to larger model errors.

To address the issue of reduced training data caused by the direct division of the labeled dataset, the strategy was revised. Instead of splitting the dataset, all labeled data remained part of the training set for model training. A subset of the data was randomly sampled and subjected to strong augmentation for use in the correction phase. This modification ensured the integrity of the training set while enhancing the diversity and representativeness of the correction set. Experiments were conducted with different λ_c, and the results are shown in Figures 2 and 3. The green line represents the performance of the FixMatch algorithm on the test set after training. The other three lines, shown in different colors, represent the results obtained with varying correction weights.

It is evident from Figures 2 and 3 that the addition of the correction mechanism increased the model's accuracy in the early training phases, particularly in the first 16 epochs. However, in the later training stages, after approximately 16 epochs, the correction capability gradually weakened. This decline is attributed to the model becoming more stable and having learned comprehensive features during the later training phases, thus diminishing the impact of further corrections.

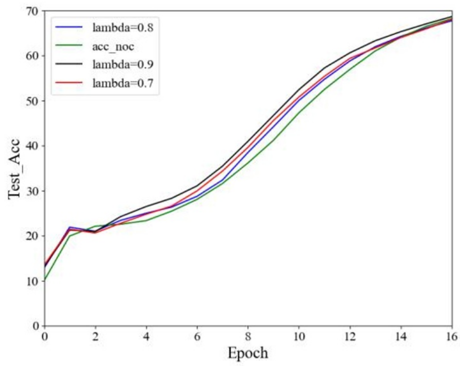

Fig. 2. This figure illustrates the results of the first 16 training epochs under different correction weights (Photo/Picture credit: Original).

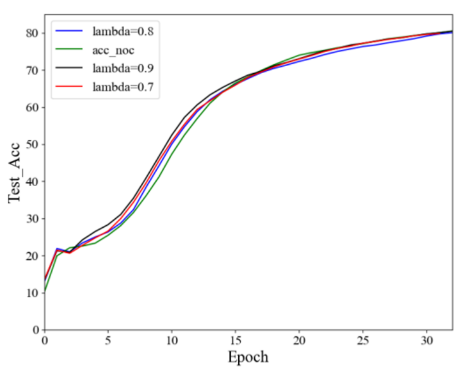

Fig. 3. Depicts the performance of different correction weights over 32 training epochs (Photo/Picture credit: Original).

During the model training and correction process described above, the correction step is by default limited to a single batch size of correction data per training step. To further enhance the model's ability to refine pseudo-labels, the size of the correction data can be increased during each correction step to include multiple batches. By concatenating these samples, a larger correction dataset is formed. The strongly augmented data is then input into the model, and the cross-entropy loss (correction loss) between the model's predictions and target labels is calculated. This correction loss is integrated into the total loss function with a specific weight, combined with the unsupervised loss to guide model optimization. Experiments were conducted using double and triple batch sizes for the correction dataset, and the results are presented in Figure 4. The green line represents the results obtained by the FixMatch algorithm after training on the test set. The blue line indicates the training results with double the correction batch size, while the black line represents the results with triple the correction batch size. The figure shows that the model achieved an accuracy of 90.29% with double correction batch sizes, which is comparable to the FixMatch baseline accuracy of 90.33%. However, with triple correction batch sizes, the test accuracy of the proposed model improved to 90.52%, demonstrating a slight enhancement compared to FixMatch.

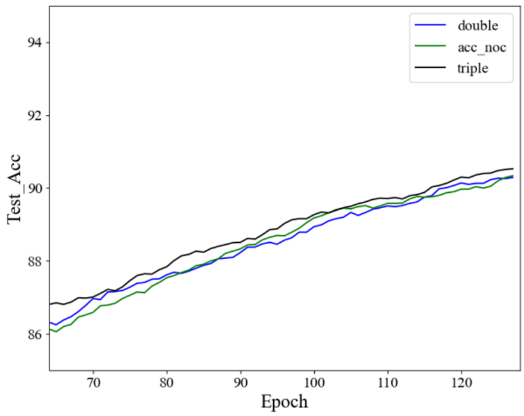

Fig. 4. This figure compares the test set accuracy results from the 64th training epoch to the end of training, under different correction batch sizes (Photo/Picture credit: Original).

4 Conclusion

Correction FixMatch demonstrates the potential of labeled data in model error correction by improving the Fixmatch model. Compared to FixMatch, Correction FixMatch performs better with triple correction batch sizes, indicating that labeled data indeed possesses a certain corrective ability. However, this experiment also has certain limitations. The study was conducted only on 4,000 labeled samples from the CIFAR-10 dataset. To establish its generalizability, extensive experiments on different datasets and varying numbers of labeled samples are needed, along with ablation studies on hyperparameters such as weight factors and correction batch sizes. Additionally, when implementing the correction mechanism, it is important to consider introducing different augmentation strategies to explore their impact on the correction process and how to apply the error correction mechanism to the entire training process.

References

1. F. M. Shiri, T. Perumal, N. Mustapha, R. Mohamed, A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU, arXiv (2024).
2. Rosenberg, M. Hebert, H. Schneiderman, Semi-supervised self-training of object detection models, In Proceedings of the Seventh IEEE Workshops on Application of Computer Vision (2005).
3. Y. Qiu, X. Gong, Z. Ma, X. Chen, MixLab: An Informative Semi-supervised Method for Multi-label Classification, In X. Zhu, M. Zhang, Y. Hong, R. He (Eds.), Natural Language Processing and Chinese Computing, 506–518, Springer International Publishing (2020).

4. Y. Wang, Z. Liu, S. Lian, Semi-supervised object detection: A survey on recent research and progress (2023).
5. Tarvainen, H. Valpola, Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results, Advances in Neural Information Processing Systems 30 (2017).
6. D. Berthelot, N. Carlini, I. J. Goodfellow, N. Papernot, A. Oliver, C. Raffel, MixMatch: A holistic approach to semi-supervised learning, CoRR, abs/1905.02249 (2019).
7. K. Sohn, D. Berthelot, N. Carlini, Z. Zhang, H. Zhang, C. A. Raffel, E. D. Cubuk, A. Kurakin, C.-L. Li, FixMatch: Simplifying semi-supervised learning with consistency and confidence, Advances in Neural Information Processing Systems 33, 596–608 (2020).
8. D. Lee, Pseudo-Label: The Simple and Efficient Semi-Supervised Learning Method for Deep Neural Networks (2013).
9. P. Bachman, O. Alsharif, D. Precup, Learning with pseudo-ensembles, In Advances in Neural Information Processing Systems, 3365–3373 (2014).
10. S. Laine, T. Aila, Temporal ensembling for semi-supervised learning, CoRR (2016).