Propagation Regularizer for Semi-supervised Learning with Extremely Scarce Labeled Samples

Noo-ri Kim Jee-Hyong Lee*
Department of Electrical and Computer Engineering, Sungkyunkwan University
2066 Seobu-ro, Jangan-gu, Suwon-si, Gyeonggi-do 16419, Republic of Korea
{pd99j, john}@skku.edu

Abstract

Semi-supervised learning (SSL) is a method to make better models using a large number of easily accessible unlabeled data along with a small number of labeled data obtained at a high cost. Most of existing SSL studies focus on the cases where sufficient amount of labeled samples are available, tens to hundreds labeled samples for each class, which still requires a lot of labeling cost. In this paper, we focus on SSL environment with extremely scarce labeled samples, only 1 or 2 labeled samples per class, where most of existing methods fail to learn. We propose a propagation regularizer which can achieve efficient and effective learning with extremely scarce labeled samples by suppressing confirmation bias. In addition, for the realistic model selection in the absence of the validation dataset, we also propose a model selection method based on our propagation regularizer. The proposed methods show 70.9%, 30.3%, and 78.9% accuracy on CIFAR-10, CIFAR-100, SVHN dataset with just one labeled sample per class, which are improved by 8.9% to 120.2% compared to the existing approaches. And our proposed methods also show good performance on a higher resolution dataset, STL-10.

1. Introduction

Semi-supervised learning (SSL) is a machine learning technique that trains a model using a small number of labeled data and a large number of unlabeled data. As it can show comparable performance to supervised learning, it is attracting more attention from researchers. Semi-supervised learning techniques have shown remarkable performances in various fields such as image segmentation [5, 8, 31], object detection [1, 11, 19], text classification [4, 10, 20], and graph embedding [27, 29] as well as image classification [17, 27].

Most SSL methods [2, 3, 12, 17, 27, 32] are based on consistency regularization [14, 26] and pseudo labeling [15, 21]. Consistency regularization is a method developed under the assumption that the prediction will not change significantly even if a slight perturbation is applied to the sample. Pseudo labeling is a special case of self-training [25, 33], which uses the predicted output of unlabeled samples as pseudo-labels to train a model.

In SSL, maximum utilization of unlabeled samples is important, but it is also important learning with a small number of labeled samples because labeled samples are usually at a high cost. However, only a few researches focused on learning with scarce labeled samples. We need to study on how SSL works and how to improve its performance in label-scarce situations.

MixMatch [3], Unsupervised Data Augmentation for Consistency Training (UDA) [32], and ReMixMatch [2] showed good performances with relatively many labeled samples such as 25, 50, 100, 200, and 400 labeled examples per class for CIFAR-10 [13]/SVHN [22] dataset. Recent approaches such as FixMatch [27], SelfMatch [12], FlexMatch [35], and CoMatch [17] considered label-scarce situations. FixMatch, SelfMatch, and FlexMatch used at least 4 labeled examples per class and CoMatch used at least 2 samples per class. However, they were unstable and showed a poor performance with a small number of labeled samples.

One of the serious problems of scarce-label situations is confirmation bias [16, 18, 30] that can occur in the label propagation [9]. Confirmation bias refers to a phenomenon in which the model learns incorrect predictions for unlabeled data, so that the confidence of the incorrect prediction is increased and the model has resistance to new (correct) information that can be corrected. If there are enough labeled data, the propagation of wrong information can be canceled out by the correct information around the incorrect prediction. SSL can avoid confirmation bias of the model. On the other hand, if labeled data is few in number, incorrect predictions can be propagated widely, and the probability of not receiving appropriate correct information can increase.

[*] Corresponding author.

Confirmation bias makes a significant adverse effect on the training of the model through SSL. This problem can be more serious in the approaches based on hard pseudo labels such as UDA, FixMatch, SelfMatch, FlexMatch.

Another serious problem in extremely label-scarce situations is the model selection. As in supervised learning, the stopping condition is very important in semi-supervised learning environments. In supervised learning, validation datasets are usually used to check the stopping condition, but in semi-supervised learning, especially in scarce-label environments, there are not enough labeled samples for validation. However, previous SSL approaches [2, 3, 12, 17, 27, 32] disregard about the stopping condition or the model selection. For performance evaluation, they simply took the median of the last 20 model performances [3, 28]. In scarce-label situations, the learning of SSL can be very unstable because of confirmation bias. A model with a low training loss does not guarantee a good test accuracy.

We propose propagation regularizer to improve the performance of SSL in extreme scarce-label environments where just 1 or 2 labeled samples are available per class. The propagation regularizer suppresses confirmation bias that can propagate incorrect predictions due to the extremely small number of labeled data, allowing SSL learning to proceed stably. We also propose a model selection method based on the loss of the propagation regularizer to select a well-trained model in extremely label-scarce scenario. These methods require very low additional computational cost and they are easy to adopt to the existing SSL approaches.

We show that confirmation bias can easily occur and make an adverse effect on model training in the extremely label-scarce scenario through toy examples and CIFAR-10 dataset. We propose propagation regularizer and the model selection method. We present the state-of-the-arts performance in an extremely label-scarce scenario with 1 or 2 labeled examples per class.

2. Confirmation Bias in Extremely Label-scarce Setting

Most SSL approaches have troubled with confirmation bias. In extreme scarce-label situation, confirmation bias problem is worsen. In this section, we evaluate how much confirmation bias makes an adverse effect on semi-supervised learning process in extreme label-scarce situations.

We conduct experiments with FixMatch [27], a representative pseudo-labeling method in SSL, and three datasets: moon dataset, star dataset and CIFAR-10 [13]. The experiments confirm that confirmation bias easily occurs in label-scarce settings and can have a significant impact on performance.

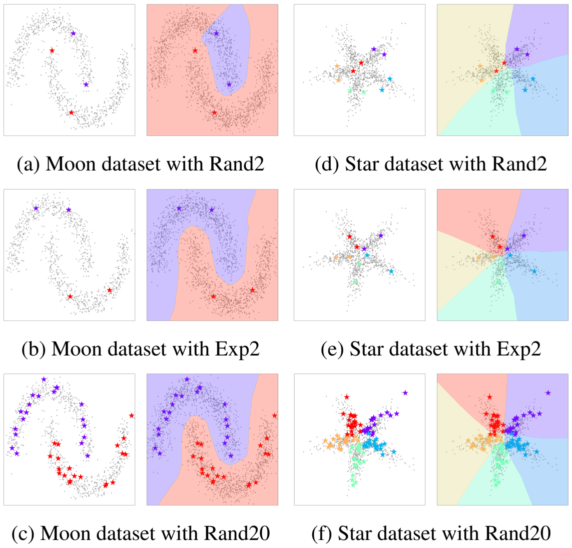

(a) Moon dataset with Rand2
(d) Star dataset with Rand2
(b) Moon dataset with Exp2
(e) Star dataset with Exp2
(c) Moon dataset with Rand20
(f) Star dataset with Rand20

Figure 1. Class boundaries by FixMatch. Labeled samples are in colors and unlabeled samples are in grey. Each crescent is a class in moon dataset and each wing is a class in star dataset.

2.1. Analysis with Toy Examples

In order to check confirmation bias occurring in SSL with extremely scarce labeled samples, we train a 3-layer neural network model with FixMatch on 2-dimensional moon and star datasets. The moon dataset consists of two classes, 1k unlabeled samples. The star dataset consists of 5 classes generated with a Gaussian distribution. Each class has 200 unlabeled samples. In each dataset, three sets of labeled examples, *Rand2*, *Exp2* and *Rand20*, are given to verity that the initial labeled samples have a significant effect on confirmation bias during training. *Rand2* contains two labeled samples per class, randomly selected from the unlabeled samples; *Exp2* consists of two labeled samples per class selected by experts so that the labeled samples represent the distribution of the unlabeled dataset well; and *Rand20* contains 20 labeled samples per class randomly selected from the unlabeled samples. For FixMatch, Gaussian noise with different strength are used for the weak and strong augmentations.

Figures 1a to 1c show the moon datasets and the learning results of FixMatch, and Figs. 1d to 1f show the result for the star datasets. It can be seen that the class boundaries do not match the data distribution when 2 randomly chosen labeled samples per class are given as shown in Figs. 1a and 1d. As shown in Figs. 1c and 1f, when more labeled samples are given, the class boundaries are properly generated. In Figs. 1b and 1e, we can notify that confirmation bias has less effect on the models if labeled samples are carefully chosen.

As shown in Figs. 1a and 1d, if labeled samples do not

Method	Fold	Class										Entropy	Accuracy
		0	1	2	3	4	5	6	7	8	9		
FixMatch	Fold 1	0.11	0.09	0.02	0.00	0.45	0.00	0.09	0.01	0.11	0.11	0.72	62.29
	Fold 2	0.11	0.10	0.02	0.08	0.26	0.00	0.10	0.09	0.10	0.13	0.90	67.18
	Fold 3	0.22	0.01	0.08	0.00	0.36	0.00	0.07	0.08	0.00	0.17	0.72	53.05
	Fold 4	0.00	0.09	0.01	0.00	0.19	0.01	0.36	0.10	0.11	0.11	0.77	51.31
	Fold 5	0.10	0.09	0.01	0.00	0.01	0.17	0.13	0.30	0.10	0.09	0.84	66.23

Table 1. Class ratio and entropy of pseudo labels for CIFAR-10 dataset with 10 labeled samples.

well represent the distribution of each class, label propagation occurs in a skewed way during SSL learning process and confirmation bias can be intensified. The label propagation process is prone to bias because there are only two labeled samples per class and the distributions of unlabeled and labeled samples do not match each other.

Even in the case where the number of labeled samples is very small, the confirmation bias can be suppressed if the labeled samples can represent the class distribution, as shown in Figs. 1b and 1e. However, it is not usually expected that a few randomly selected samples properly represent the data distribution. As seen in Figs. 1c and 1f, if there are many randomly chosen labeled samples, they can represent the class distribution, and the class boundaries are learned properly.

2.2. Analysis with CIFAR-10 Dataset

In order to verify that real-world datasets are prone to the confirmation bias problem, we conduct the experiment with the CIFAR-10 dataset. In this experiment, we use 1 labeled sample per class and train FixMatch with Wide Residual Network 28-2 model [34]. The experiment was performed in 5 folds and, in each fold, labeled examples are randomly selected from the training data.

The performance is the median accuracy of the last 20 models, and it is averaged over 5 folds. The average accuracy is 60.01%. The highest accuracy among 5 folds is 67.18%, and the lowest is 51.31%, showing a large variance in performance.

In order to prove that each model of 5 folds does not generate a good model due to confirmation bias, we observe the class ratio of pseudo-labels in Tab. 1. If each model is well trained without confirmation bias on the CIFAR-10 dataset, the ratio of each class in pseudo-labels will appear as 0.1, and the entropy of the ratios will be 1.0. The entropy is defined as follows:

$$Entropy = -\sum_{i}^{c} r_i \log_c r_i \tag{1}$$

where r_i is the ratio of class i and c is the number of classes.

We notice that the class ratios in each fold are not balanced in Tab. 1. In detail, in the first fold, unlabeled samples are never pseudo-labeled as Class 3 or 5 and only small number of unlabeled samples are pseudo-labeled as Class 7, which results in a low entropy of 0.72. Such tendency is observed for all 5-fold. And even in Fold 2 with the highest entropy of 0.90, the ratio of Class 3 is 0. The average of 5-fold entropy for pseudo-labeling of FixMatch on CIFAR-10 with 10 labeled samples is 0.79. What is interesting is that the entropy and the accuracy of the model has a strong correlation of 0.69. Higher entropy means smaller confirmation bias; thus, we surmise that confirmation bias has a significant effect on a model's performance. We also run the same experiment with 25 labeled samples per class. The average entropy is 0.99, which means that there is little confirmation bias.

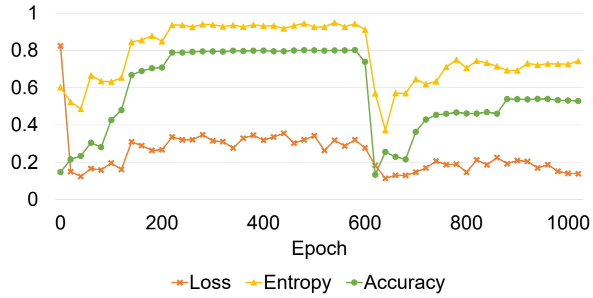


Figure 2. Training loss, entropy of pseudo-labels and test accuracy of FixMatch on CIFAR-10 with 10 labeled samples (Fold 3).

Through the experiment, we confirm that confirmation bias is also easy to occur in real-world datasets, that the smaller the number of labeled samples, the stronger effect confirmation bias makes, and that the strength of confirmation bias, measured as entropy of pseudo-class ratios, is strongly associated with the model performance.

We also observe the test accuracy, the training loss, and the entropy of pseudo-labels by epoch. Figure 2 shows the accuracy, the training loss, and the entropy of Fold 3 by epoch. We see that the training is very unstable. At the beginning of training, the entropy of pseudo-labels increases, and the test accuracy also increases. This shows that consistency regularization performs beneficially along with pseudo-labeling and the SSL model is being trained well. However, the test accuracy drops sharply around 600

Pearson's Correlation Coefficient	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Average
Training Loss-Accuracy	0.175	0.364	0.747	0.257	0.406	0.390
Entropy-Accuracy	0.835	0.807	0.950	0.841	0.846	0.856

Table 2. Pearson's Correlation Coefficient of training loss-test accuracy and entropy-test accuracy during FixMatch training on the CIFAR-10 with 10 labeled samples.

epochs and the same time the entropy also sharply drops. Afterward, the performance of the model recovers to some extent, but it hardly regains the previous best performance.

From the observation, we notice two things. The performance of the model does not gradually improve as the training progresses. The learning of the model is unstable, showing the best performance in the middle of training, and then rapidly dropping at some point. Therefore, choosing the last updated model or the model with the lowest training loss does not guarantee the best model.

The second is more important. We perceive that the correlation of test accuracy and pseudo-label class entropy is much higher than that of test accuracy and training loss. Table 2 shows the correlation coefficients between test accuracy and entropy, and between test accuracy and training loss. This observation gives us a hint on how to suppress confirmation bias in training and how to better select a good model.

3. Proposed Method

In pseudo-labeling process, the model learns the model output, i.e., it repeatedly learns its own erroneous prediction, resulting in confirmation bias [18, 30]. This phenomenon can be amplified especially in an extremely scarce labeled scenario with one or two labeled examples in each class.

Based on our experimental observations, we propose a propagation regularizer method to suppress confirmation bias in an extremely scarce label environment, and a model selection method that selects the optimal model without validation data among models generated during the learning process.

3.1. Propagation Regularizer

In pseudo-labeling process, incorrect predictions of the model can be used for the next model training, which causes confirmation bias. We may infer that we need to keep the balance between pseudo-labels based on our observation that the correlation between test accuracy and the entropy of pseudo-classes is high as shown in Tab. 2. Learning imbalanced pseudo-labeled sample will augment confirmation bias.

For example, let us consider SSL learning with two classes, A and B. If a model in the middle of SSL training produces more pseudo-labels of class A than B, the imbalanced pseudo-labeled samples are used for the next model training. Then, the next model is easy to be biased to class A, and the confirmation bias will be inflated.

To solve this problem, a regularization term is designed so that the pseudo-labeling for the unlabeled samples should be balanced for each class as follows:

$$L_{pr} = 1 - (-\mathbf{P}_{\mathrm{U}} \cdot \log_c(\mathbf{P}_{\mathrm{U}})) \tag{2}$$

where c is the number of classes. $\mathbf{P}_{\mathrm{U}}$ is the masked averaged probability distribution of unlabeled examples, unlabeled samples U in a batch, defined as follows:

$$\mathbf{P}_{\mathrm{U}} = \frac{1}{|\mathrm{U}|}\sum_{u\in\mathrm{U}} \mathbb{1}\left(\max\left(p\left(u\right)\right) \geq \tau\right) p\left(u\right) \tag{3}$$

where τ is the confidence threshold for pseudo-labeling and $p(u)$ is the softmax output of an unlabeled example u.

In Eq. (3), the average of predictions is obtained for samples having values greater than or equal to a threshold τ in a batch of unlabeled examples. To convert this to a minimization form, the entropy of $\mathbf{P}_{\mathrm{U}}$ is subtracted from 1. If the pseudo-labels of unlabeled examples are evenly distributed, the value of L_{pr} will converge to 0. By simply adding this regularization term to the SSL loss, class-balanced pseudo-labeling can be achieved. Through this, we can alleviate the confirmation bias in extremely scarce example scenario.

3.2. Model Selection based on Propagation Regularizer and Utilization

Model selection is crucial in semi-supervised learning. As observed in Sec. 2, the performance of the model is not stable during training, because it is affected much by confirmation bias. If we have a validation dataset, we may choose the best model as we do in supervised learning. However, there are not sufficient labeled samples to be used for validation in our case.

Some SSL approaches [14, 21, 24, 30, 32] simply select the last model. Many recent SSL studies [2,3,12,17,27,32] did not propose a model selection method. They took the median of the performance of the last 20 models for model evaluation. This may be acceptable as a model performance comparison method [3,28] in plain environments. However, as shown in Fig. 2, model performance is very unstable in an extreme label-scarce environment. This makes model

Method	Fold	Class										Entropy	Accuracy
		0	1	2	3	4	5	6	7	8	9		
FixMatch **+ Sel + Reg**	Fold 1	0.10	0.12	0.07	0.08	0.18	0.11	0.12	0.03	0.11	0.09	0.97	68.60
	Fold 2	0.09	0.12	0.06	0.06	0.09	0.09	0.18	0.10	0.10	0.11	0.98	59.47
	Fold 3	0.07	0.17	0.04	0.09	0.13	0.08	0.12	0.03	0.09	0.17	0.95	72.89
	Fold 4	0.17	0.11	0.02	0.05	0.10	0.15	0.13	0.07	0.12	0.08	0.95	78.81
	Fold 5	0.12	0.11	0.01	0.10	0.03	0.10	0.12	0.21	0.12	0.08	0.94	74.58

Table 3. Class ratio and entropy of pseudo labels for CIFAR-10 dataset with 10 labeled samples. The proposed model selection and propagation regularizer are applied.

selection very difficult and hinders SSL approaches from being used in real-world applications.

To select an appropriate model, we propose a measure based on confirmation bias and utilization of unlabeled samples. A good SSL model would utilize unlabeled samples as much as possible and be less affected by confirmation bias. To choose such models, we propose the utilization measure of unlabeled samples and the influence measure of confirmation bias.

For the utilization measure of unlabeled samples, we propose the following equation:

$$T_{\mathrm{U}} = \frac{1}{|\mathrm{U}|}\sum_{u\in\mathrm{U}} \mathbb{1}(\max(p(u)) \geq \tau) \tag{4}$$

Equation (4) shows the ratio of pseudo-labeled examples satisfying the confidence threshold for pseudo-labeling, τ. If the model uses all unlabeled examples in a batch for training, the value of T_{U} is 1; and if none of the unlabeled examples is used at all, it is 0. To measure the influence of confirmation bias, we use Eq. (2), which is the proposed propagation regularizer. By combining Eqs. (2) and (4), we develop a metric for the model selection. It is defined as follows:

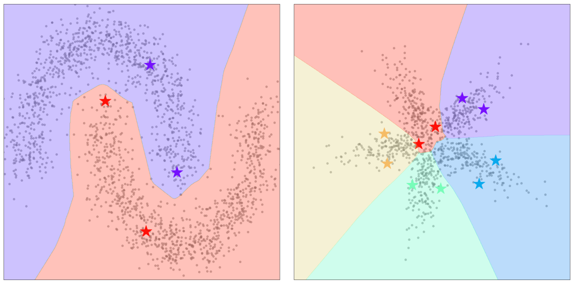
$$Sel = (1 - L_{pr}) + T_{\mathrm{U}} \tag{5}$$

A good SSL model utilizes most unlabeled samples and is less affected by confirmation bias, so the value will be maximized. In the training process, we evaluate Sel at each epoch, and choose the model with the maximum value of Sel as the final model.

The proposed model selection method does not use an additional validation dataset. We can select an appropriate SSL model without a validation dataset in scarce-label situations.

4. Experiment

To verify the proposed propagation regularizer and model selection methods, we combine the proposed methods to each of UDA [32] and FixMatch [27], and we perform SSL image classification benchmarks. We compare the performance with the current SOTA approaches, CoMatch [17] and FlexMatch [35], with SVHN [22], CIFAR-10 and CIFAR-100 [13]. Also, we conduct experiments on a higher resolution dataset, STL-10 [6], for FixMatch with our proposed methods. We perform the SSL methods on the datasets with a various number of labeled examples including extremely label-scare scenarios. All experiments were performed according to SSL evaluation protocols [2, 3, 23]. The experiment results show the superiority of the proposed method. Our methods show the best performance in the extremely label-scarce scenario.



(a) Moon dataset with Rand2

(b) Star dataset with Rand2

Figure 3. Datasets and class boundaries by FixMatch with the proposed method. In the dataset, labeled samples are in colors unlabeled samples are in grey. In moon datasets, each crescent is a class and in star datasets, each wing is a class.

4.1. Propagation Regularizer with Toy Examples and CIFAR-10 Dataset

To confirm that our proposed propagation regularizer works effectively, we apply the proposed method to the experiment in Sec. 2.

The experimental results for the moon and star dataset are shown in Fig. 3. In Figs. 1a and 1d, the learned class distribution are not well represented the data class distribution because confirmation bias can be intensified. When the proposed propagation regularizer is applied to FixMatch, it can be seen that the class distribution is properly learned as Figs. 3a and 3b.

Table 3 shows the class ratio of pseudo-labels of unlabeled examples, entropy, and accuracy when FixMatch with

Method	CIFAR-10			CIFAR-100			SVHN		
	10 labels	20 labels	40 labels	100 labels	200 labels	400 labels	10 labels	20 labels	40 labels
UDA	51.82 ±8.51	83.53 ±8.05	90.06 ±4.37	24.96 ±2.22	37.76 ±0.74	48.98 ±1.73	24.02 ±19.68	67.84 ±11.05	96.51 ±2.54
FixMatch	60.01 ±7.41	75.50 ±10.93	85.57 ±5.21	24.11 ±1.50	35.83 ±1.63	46.04 ±1.41	35.84 ±10.12	55.79 ±30.83	86.73 ±21.70
CoMatch	65.10 ±7.81	88.26 ±8.29	92.16 ±4.97	24.19 ±0.98	32.51 ±1.15	41.72 ±2.04	25.36 ±4.64	45.62 ±7.12	76.07 ±9.31
FlexMatch	59.06 ±19.80	**94.62** ±0.15	**94.86** ±0.05	4.47 ±0.81	30.59 ±1.69	46.11 ±2.83	11.02 ±1.89	34.93 ±36.00	77.04 ±23.16
UDA **+ Sel**	59.71 ±16.01	83.60 ±8.09	90.12 ±4.35	24.95 ±2.23	37.76 ±0.84	49.11 ±1.77	**78.91** ±12.30	**97.78** ±0.27	96.48 ±3.24
UDA **+ Sel + Reg**	69.87 ±9.96	84.33 ±7.23	91.54 ±2.53	**30.30** ±0.99	**42.34** ±1.54	**50.61** ±1.48	77.98 ±32.06	96.01 ±3.71	97.46 ±0.45
FixMatch **+ Sel**	65.73 ±10.32	79.24 ±10.00	89.87 ±4.96	24.17 ±1.55	35.78 ±1.58	46.05 ±1.28	51.40 ±26.66	91.90 ±5.77	96.41 ±3.06
FixMatch **+ Sel + Reg**	**70.87** ±7.35	88.20 ±4.29	91.52 ±2.81	27.97 ±1.12	38.96 ±1.42	48.01 ±1.72	69.61 ±24.33	96.26 ±2.86	**97.61** ±0.33

Table 4. Comparison of accuracy for CIFAR-10, CIFAR-100 and SVHN on 5 different folds with 1, 2 and 4 labeled samples per class.

Method	STL-10		
	10 labels	20 labels	40 labels
FixMatch	30.82 ±6.73	43.24 ±6.32	60.92 ±5.60
FixMatch **+ Sel**	30.07 ±5.82	45.45 ±3.24	63.93 ±9.65
FixMatch **+ Sel + Reg**	**37.91** ±6.66	**61.00** ±15.87	**74.45** ±13.50

Table 5. Comparison of accuracy for STL-10 on 5 different folds with 1, 2 and 4 labeled samples per class.

the proposed method is applied to the CIFAR-10 dataset. In Tab. 3, the class ratios in each fold are more balanced than in Tab. 1, and the accuracy is also improved. The average of entropy increases from 0.79 to 0.96, and the average performance increases from 60.01% to 70.87%. It shows that the proposed method is working effectively in extremely label-scarce scenario.

4.2. Dataset and Implementation Details

We conduct experiments on CIFAR-10, CIFAR-100, SVHN and STL-10. CIFAR-10/100 and SVHN datasets consist of 3 channels of 32×32 size, and STL-10 consists of 3 channels of $96\times\ 96$. CIFAR-10,SVHN and STL-10 consist of 10 classes, and CIFAR-100 consists of 100 classes. CIFAR-10 consists of 50,000 training images and 10,000 test images. CIFAR-100 consists of 60,000 training images and 10,000 test images. SVHN consists of 73,257 training images, 26,302 test images, and 531,131 additional images. STL-10 consists of 5,000 training images and 100,000 unlabeled images, and 8,000 test images. In CIFAR-10 and CIFAR-100, images to be used as labeled data are randomly selected class-evenly from the training images and the remaining training images are treated as unlabeled data. In SVHN and STL-10, images to be used labeled data are chosen as the same way from the training and additional images, and the remainders are treated as unlabeled data. Unlike CIFAR-10/100 and STL-10, SVHN is not class balanced dataset. The number of samples for each class varies from 6.47% to 17.28% of the total data.

We set λ_U = 1, η = 0.03, β = 0.9, τ = 0.95, μ = 7, and B = 64 for FixMatch, and λ_{cls} = 1, η = 0.03, τ = 0.95, μ = 7, B = 64, α = 0.9, τ = 0.2, K = 2560, T = 0.8, and λ_{ctr} = 1 for CoMatch. Those hyperparameters are set based on the original works [17, 27]. For UDA, we adopt the same values used by Sohn et al. [27]: λ_U = 1, η = 0.03, temperature τ = 1, confidence threshold β = 0.9, μ = 7, and B = 64. We use RandAugment [7] as strong augmentation for CoMatch and CTAugment [2] for FixMatch and UDA. The weight factor of the propagation regularizer, L_{pr}, is set 1.0 for CIFAR-100, and 0.4 for CIFAR-10, SVHN, and STL-10. We use a Wide ResNet-28-2 [34] for CIFAR-10/100 and SVHN, and Wide ResNet-37-2 for STL-10.

4.3. Results for Extremely Label-scarce Scenario

Results for extremely label-scarce scenario are shown in Tab. 4. The baselines are UDA, FixMatch and CoMatch. MixMatch and ReMixMatch were excluded from the baselines because of low performance with extremely label-scarce scenario. Their performance of CIFAR-10

Method	CIFAR-10			CIFAR-100			SVHN		
	40 labels	100 labels	250 labels	400 labels	1000 labels	2500 labels	40 labels	100 labels	250 labels
FixMatch	85.57 ±5.21	92.63 ±3.21	**95.08** ±0.08	46.04 ±1.41	57.90 ±1.21	64.78 ±0.48	86.73 ±21.70	97.78 ±0.20	**97.98** ±0.21
FixMatch **+ Sel**	89.87 ±4.96	93.72 ±1.14	94.13 ±0.69	46.05 ±1.28	**57.96** ±1.04	64.71 ±0.35	96.41 ±3.06	**97.81** ±0.15	97.95 ±0.21
FixMatch **+ Sel + Reg**	**91.52** ±2.81	**94.02** ±0.98	94.43 ±0.62	**48.01** ±1.72	57.63 ±0.96	**65.21** ±0.49	**97.61** ±0.33	97.73 ±0.44	97.69 ±0.56

Table 6. Comparison of accuracy for CIFAR-10, CIFAR-100 and SVHN on 5 different folds with 4, 10 and 25 labeled samples per class.

with 10 labeled samples was 17.48% and 31.00%, respectively, which are very lower than the other baselines. Our approaches are applied to UDA and FixMatch. For example, FixMatch+Sel is the performance of the model trained by FixMatch with our model selection, and FixMatch+Sel+Reg is for the model trained by FixMatch with our propagation regularizer and model selection. We evaluate the cases where the number of labeled samples per class is 1, 2, and 4. The baseline approaches did not propose how to select trained models, we evaluate them as the authors did. We choose the median of the last 20 models.

The methods combined with the proposed propagation regularizer and model selection show the best performance in every case except CIFAR-10 with 20 and 40 labeled samples.

In CIFAR-10 with 10 labeled samples, the accuracy of FixMatch+Sel+Reg is 70.87%, which shows an 8.9% improvement compared to CoMatch. With 20 and 40 labels, CoMatch performs slightly better than models with our approaches. However, the variance of CoMatch is almost twice of ours the best model. In 20 labels, the variance of CoMatch is 8.29 but that of FixMatch+Sel+Reg is 4.29, and they are 4.97 and 2.81 in 40 labels, respectively. FlexMatch shows best performance with 20 and 40 labels, but the performance drops sharply with 10 labels, showing the second-worst performance.

The experiments with CIFAR-100 also interesting results. UDA+Sel+Reg is the best with 100, 200, and 400 labels. The improvements over the best baselines are 21.4%, 12.1% and 3.3%, respectively. These results show the effectiveness of our approaches in large datasets.

The performance improvements with SVHN are 120.2%, 44.1% and 1.1% for 10, 20 and 40 labels, respectively. In the most scarce case where there is 1 labeled sample per class, our approaches improve the most. The proposed regularization and model selection methods effectively worked on class imbalanced datasets, such as SVHN, as well as on class balanced datasets, such as CIFAR-10 and CIFAR-100.

Our method most improves the performance on SVHN than other class-balanced datasets. Our method maintains the class balance of pseudo-labels even in imbalanced datasets. Performance can be improved by preventing the confirmation bias in the early stage of learning that may occur due to the class imbalance. It helps suppressing confirmation bias as well as dealing with class imbalance.

Especially, our approaches improve much with 1 labeled sample per class. In CIFAR-10 with 10 labeled samples, the improvements of UDA+Sel and UDA+Sel+Reg over UDA are 15.2% and 34.8%, respectively, and the improvements of FixMatch are 9.5% and 18.1%, respectively. In SVHN with 10 labels, the improvements of UDA are 228.5% and 224.6%, and those are 43.4% and 94.2% for FixMatch.

FlexMatch shows the best performance in CIFAR-10 with 20 and 40 labeled samples, but it shows the worst or the almost worst performance in the other cases. CoMatch shows similar performance patterns to FlexMatch. FlexMatch and CoMatch showed good performances with enough labeled samples, but they shows bad performances in extremely scarce label scenarios.

We also perform experiments with higher resolution datasets, STL-10, which has 96 × 96 images. Table 5 shows the experimental results. FixMatch+Sel+Reg shows 23%, 39.8%, and 22.2% performance improvement over FixMatch, confirming that our methods are also effective in STL-10 dataset.

Through the experiments, we verify that our propagation regularizer and model selection are very effective to improve the performance of SSL approaches in extremely label-scarce scenarios.

4.4. Results with More Examples

Table 6 shows the performance of the propagation regularization and model selection methods for FixMatch with more labeled examples. We can notice that our approaches are still valid with large labeled datasets.

In the cases with 10 and 25 labeled samples per class, the performance gain is reduced because confirmation bias will also reduce if there are many labeled samples. When the number of labeled samples is large enough, confirmation bias can be easily suppressed and the model can be trained stably. Also, in an environment with enough labeled examples, the propagation regularizer will lost its influence. Its value will be zero because there is little confirmation bias.

Through these experiments, we can conclude that our approaches effectively suppress confirmation bias regardless of the number of labeled samples and are generally applicable.

4.5. Computational Cost

Since the proposed model selection method is performed after training, the training time does not increase. The proposed propagation regularizer is performed on each training batch, but it requires only a very small computation cost. When comparing the actual training time, it increases only by 0.64%. This makes practically no difference.

5. Related Work

Semi-supervised learning tries to make better models using a large number of easily accessible unlabeled data along with a small number of labeled data obtained at a high cost.

In studies based on consistency regularization methods, which are one of the representative methods of semi-supervised learning, there are many studies such as π model [14], temporal ensembling [14], Mean Teacher [30] and VAT [21]. These studies have been conducted using hundreds to thousands of labeled data. After that, with the continuous development of SSL studies, its performance has been improved, and the number of labeled samples required for semi-supervised learning has been reduced. UDA [32] and ReMixMatch [2], which use a method of applying consistency regularization, strong augmentation, and sharpening to pseudo-labels, showed good performance by performing semi-supervised learning with fewer samples per class than previous studies.

Recently, FixMatch [27] and various studies based on it are being conducted [12, 17, 35]. FixMatch has a relatively simple structure based on UDA and ReMixMatch. Consistency training is performed by applying strong augmentation such as CTAugment [2], and pseudo-labeling with threshold. Unlike the relatively simple structure, it showed outperformed performance on various dataset with a few labeled examples such as 4 label examples per class. FlexMatch [35] used flexibly adjust thresholds for different classes to consider different learning status and learning difficulties of different classes. SelfMatch [12] and CoMatch [17] are SSL methods that adopt self-supervised learning. SelfMatch showed that self-supervised learning can serve rich information for SSL model initialization. CoMatch jointly learns class probabilities and embeddings, and adopt memory-smoothed pseudo-labeling to mitigate confirmation bias. FixMatch, FlexMatch, SelfMatch, and CoMatch showed good performance on small number of samples. However, it is necessary to understand the problems that can occur in an extremely label-scarce environment. Also, most SSL studies are not focused on how to select better models on training. But Model selection methods need to be studied in order to be applied to various applications in the real-world.

6. Conclusion

Many SSL approaches have been proposed, but they still show low accuracy and instability in the extremely label-scarce scenario.

We confirmed that confirmation bias had a serious impact on performance degradation through experiments in an environment with extremely small labeled data. We proposed a propagation regularizer which led to efficient and effective learning with extremely scarce labeled samples by suppressing confirmation bias. We also proposed a model selection method without the validation dataset based on our propagation regularizer.

Our methods are easy to adapt to the existing SSL methods and showed high performance and stability, even though it required very low additional computational cost.

Acknowledgement

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-00421, Artificial Intelligence Graduate School Program (Sungkyunkwan University)). Also, this research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ICT Creative Consilience program (IITP-2021-2020-0-01821) supervised by the IITP (Institute for Information & communications Technology Planning & Evaluation).

References

[1] Amir Bar, Xin Wang, Vadim Kantorov, Colorado Reed, Roei Herzig, Gal Chechik, Anna Rohrbach, Trevor Darrell, and A. Globerson. Detreg: Unsupervised pretraining with region priors for object detection. *ArXiv preprint*, abs/2106.04550, 2021. 1

[2] David Berthelot, Nicholas Carlini, Ekin D. Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. 1, 2, 4, 5, 6, 8

[3] David Berthelot, Nicholas Carlini, Ian J. Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5050–5060, 2019. 1, 2, 4, 5

[4] Jiaao Chen, Zichao Yang, and Diyi Yang. MixText: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online, 2020. Association for Computational Linguistics. 1

[5] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, 2021. 1

[6] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223. JMLR Workshop and Conference Proceedings, 2011. 5

[7] Ekin Dogus Cubuk, Barret Zoph, Jon Shlens, and Quoc Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 6

[8] Geoffrey French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham D. Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. In *31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020*. BMVA Press, 2020. 1

[9] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 5070–5079. Computer Vision Foundation / IEEE, 2019. 1

[10] Pavel Izmailov, Polina Kirichenko, Marc Finzi, and Andrew Gordon Wilson. Semi-supervised learning with normalizing flows. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 4615–4630. PMLR, 2020. 1

[11] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 10758–10767, 2019. 1

[12] Byoungjip Kim, Jinho Choo, Yeong-Dae Kwon, Seongho Joe, Seungjai Min, and Youngjune Gwon. Selfmatch: Combining contrastive self-supervision and consistency for semi-supervised learning. *ArXiv preprint*, abs/2101.06480, 2021. 1, 2, 4, 8

[13] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 1, 2, 5

[14] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. 1, 4, 8

[15] Dong-Hyun Lee. Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks. *ICML 2013 Workshop : Challenges in Representation Learning (WREPL)*, 07 2013. 1

[16] Hye-Woo Lee, Noo-ri Kim, and Jee-Hyong Lee. Deep neural network self-training based on unsupervised learning and dropout. *International Journal of Fuzzy Logic and Intelligent Systems*, 17(1):1–9, 2017. 1

[17] Junnan Li, Caiming Xiong, and S. Hoi. Comatch: Semi-supervised learning with contrastive graph regularization. *ArXiv preprint*, abs/2011.11183, 2020. 1, 2, 4, 5, 6, 8

[18] Lu Liu, Yiting Li, and R. Tan. Decoupled certainty-driven consistency loss for semi-supervised learning. *ArXiv preprint*, abs/, 20. 1, 4

[19] Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, B. Wu, Z. Kira, and Péter Vajda. Unbiased teacher for semi-supervised object detection. *ArXiv*, abs/2102.09480, 2021. 1

[20] Takeru Miyato, Andrew M. Dai, and I. Goodfellow. Adversarial training methods for semi-supervised text classification. *arXiv: Machine Learning*, 2017. 1

[21] Takeru Miyato, S. Maeda, Masanori Koyama, and S. Ishii. Virtual adversarial training: A regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:1979–1993, 2019. 1, 4, 8

[22] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011. 1, 5

[23] Avital Oliver, Augustus Odena, Colin Raffel, Ekin Dogus Cubuk, and Ian J. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 3239–3250, 2018. 5

[24] Hieu H. Pham, Qizhe Xie, Zihang Dai, and Quoc V. Le. Meta pseudo labels. In *CVPR*, 2021. 4

[25] C. Rosenberg, M. Hebert, and H. Schneiderman. Semi-supervised self-training of object detection models. *2005 Seventh IEEE Workshops on Applications of Computer Vision (WACV/MOTION'05) - Volume 1*, 1:29–36, 2005. 1

[26] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1163–1171, 2016. 1

[27] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems*, volume 33, pages 596–608. Curran Associates, Inc., 2020. 1, 2, 4, 5, 6, 8

[28] J. Su, Zezhou Cheng, and Subhransu Maji. A realistic evaluation of semi-supervised learning for fine-grained classification. In *CVPR*, 2021. 2, 4

[29] Fan-Yun Sun, Jordan Hoffmann, Vikas Verma, and Jian Tang. Infograph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. 1

[30] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1195–1204, 2017. 1, 4, 8

[31] Peng Tu, Yawen Huang, Rongrong Ji, Feng Zheng, and L. Shao. Guidedmix-net: Learning to improve pseudo masks using labeled images as reference. *ArXiv preprint*, abs/2106.15064, 2021. 1

[32] Qizhe Xie, Zihang Dai, E. Hovy, Minh-Thang Luong, and Quoc V. Le. Unsupervised data augmentation for consistency training. *ArXiv preprint*, abs/, 20. 1, 2, 4, 5, 8

[33] Qizhe Xie, Minh-Thang Luong, Eduard H. Hovy, and Quoc V. Le. Self-training with noisy student improves imagenet classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 10684–10695. IEEE, 2020. 1

[34] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016*. BMVA Press, 2016. 3, 6

[35] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 5, 8