This CVPR workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Contrastive Regularization for Semi-Supervised Learning

Doyup Lee
POSTECH, Kakao Brain
doyup.lee@postech.ac.kr

Sungwoong Kim
Kakao Brain
swkim@kakaobrain.com

Ildoo Kim
Kakao Brain
ildoo.kim@kakaobrain.com

Yeongjae Cheon
Kakao Enterprise
yj.one@kakaoenterprise.com

Minsu Cho
POSTECH
mscho@postech.ac.kr

Wook-Shin Han *
POSTECH
wshan@postech.ac.kr

Abstract

Consistency regularization on label predictions becomes a fundamental technique in semi-supervised learning, but it still requires a large number of training iterations for high performance. In this study, we analyze that the consistency regularization restricts the propagation of labeling information due to the exclusion of samples with unconfident pseudo-labels in the model updates. Then, we propose contrastive regularization to improve both efficiency and accuracy of the consistency regularization by well-clustered features of unlabeled data. In specific, after strongly augmented samples are assigned to clusters by their pseudo-labels, our contrastive regularization updates the model so that the features with confident pseudo-labels aggregate the features in the same cluster, while pushing away features in different clusters. As a result, the information of confident pseudo-labels can be effectively propagated into more unlabeled samples during training by the well-clustered features. On benchmarks of semi-supervised learning tasks, our contrastive regularization improves the previous consistency-based methods and achieves state-of-the-art results, especially with fewer training iterations. Our method also shows robust performance on open-set semi-supervised learning where unlabeled data includes out-of-distribution samples.

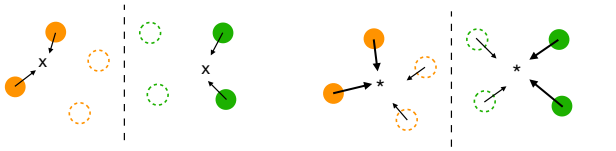
1. Introduction

Recent semi-supervised learning (SSL) methods mostly make use of the consistency regularization to learn a specific task with sparse labels, showing competitive results to the fully supervised learning [3, 20, 28]. The consistency regularization enforces a model to produce consistent predictions on various augmented views of input with pseudo-labeling [19]. Moreover, in order to avoid a confirmation bias [1] and increase the reliability of pseudo-labeling, a selection mask is typically used in this consistency regularization to exclude unconfident label predictions during SSL training. Consequently, the consistency regularization can propagate the labeling information into unlabeled samples around the augmented views of confident pseudo-labels [11].

Despite its promising results, the existing consistency regularization requires an expensive training cost to achieve high performance. For example, although FixMatch [28] can achieve high SSL performance without pretraining on a large scale unlabeled data [8], it needs over 10,000 epochs to obtain the best performance even on small-scale datasets such as SVHN, CIFAR-10, or CIFAR-100. Thus, we first analyze the inefficiency of the consistency regularization for SSL, both theoretically and empirically, and then verify that this inefficiency is originated from the exclusion of samples with unconfident pseudo-labels when updating a model. Namely, it restricts the active propagation of confident labeling information into unlabeled samples, especially in the early stage of training.

Based on the above analysis, we propose *contrastive regularization* to improve the performance of SSL based on consistency regularization. The main idea is described in Figure 1. The consistency regularization moves the features of strongly augmented samples having only confident pseudo-labels toward their corresponding class centers of the confident features by pseudo-labels. In contrast, the proposed contrastive regularization forms class clusters based on both confident and unconfident pseudo-labels. Then, it moves the features having confident pseudo-labels toward the center positions of their clusters, while pulling the features of samples with both confident and unconfident pseudo-labels in the same cluster and pushing the features in different clusters. Thus, a model can learn well-clustered features of unlabeled data, enabling the confident labeling

*Corresponding author



(a) Consistency Regularization (b) Contrastive Regularization (ours)

Figure 1. The feature update of (a) consistency and (b) contrastive regularization. Different colors represent pseudo-labels. The circles with solid and dashed line are penultimate features having confident and unconfident pseudo-labels, respectively. The black dashed line is the decision boundary. The symbol $\times$ represents a cluster center that is estimated by confident samples only, and * represents a cluster center that is estimated by all samples in the same cluster. The length of arrows represents the magnitude of gradient vectors. The cluster centers are computed by the class weight vectors.

information to be propagated into more unlabeled samples during training.

In the experiments, we show that our contrastive regularization improves the performance of consistency regularization methods on various SSL benchmarks, including SVHN, CIFAR-10, CIFAR-100, STL-10, and ImageNet with limited labels. Especially, different from the previous methods, we show that our method leverages unlabeled samples in the early stage of training and requires much fewer iterations for the outperformance. We also demonstrate that the contrastive regularization achieves the robust performance on the task of open-set SSL, which is more realistic in that unlabeled data contains out-of-distribution samples [22, 33]. Finally, we conduct an extensive ablation study to show that the contrastive regularization is valid and not highly sensitive to the selection of hyper-parameters.

Our main contributions can be summarized as follows. 1) It is the first study to analyze the limitation of the consistency regularization with respect to the efficiency of SSL training. 2) We propose a simple yet powerful solution, the contrastive regularization, which consistently improves the SSL performance on different SSL benchmarks with fewer training iterations than the previous consistency regularization. 3) Contrastive regularization shows the robustness on the more realistic benchmark that includes out-of-distribution samples in the unlabeled dataset.

2. Related Work

Consistency Regularization for SSL. Recent SSL methods use the consistency regularization [26] and focus on the policies of stochastic data augmentations such as adversarial perturbations [21] or mixup [3, 4, 37]. As the most simplified yet powerful framework of SSL, UDA and FixMatch [28, 32] show that the simple combination of strong data augmentation such as RandAugment [10] and pseudo-labeling [19] can obtain high performance. Thus, we focus on improving the consistency regularization, because they have shown state-of-the-art results compared with other SSL approaches [14, 27].

Semi-Supervised Learning with Self-Supervision. The SSL performance can be improved when self-supervised learning is used with an auxiliary task for representation learning, and our contrastive regularization can be viewed as an auxiliary task for SSL. S4L [36] demonstrates that auxiliary tasks such as rotation or exemplar self-supervision can improve the SSL performance. For time-series classification, forecasting of the next-step value is used as an auxiliary task [15]. CoMatch [20] unifies pseudo-labeling, self-supervised learning, and graph-based SSL, using the graph contrastive learning and the pseudo-label smoothing with a large size of memory bank [13]. Although both CoMatch and our method use a contrastive loss to regularize the unlabeled features, our method can be easily in tandem with the consistency regularization with minimal change for the contrastive loss.

Pretraining and Finetuning. Finetuning after pretraining on an upstream task is a solution for learning a task with scarce labels, when large-scale labeled or unlabeled datasets are available for the upstream task. For instance, a model, which is pretrained on a large-scale labeled dataset, can be well transferred to downstream tasks [18]. but a negative transfer occurs when the target task is unrelated to the upstream domain or task [35]. When a large-scale unlabeled data is available, a framework using both *task-agnostic* pretraining and task-specific finetuning can become a strong SSL approach [8, 13, 31]. However, utilizing unlabeled samples in a task-specific way can outperform a task-agnostic approach without a large number of unlabeled samples. Thus, we emphasize that *task-specific* SSL methods are important because it is hard to collect a large number of unlabeled samples in the real world.

3. Contrastive Regularization for Semi-Supervised Learning

In this section, we introduce our *contrastive regularization* to improve the SSL performance of the consistency regularization. We first formulate SSL and the consistency regularization, which is the most common approach and shows remarkable results with deep neural networks (DNNs).

3.1. Problem Formulation

We assume that a labeled dataset D_L and an unlabeled dataset D_U are given to train a model parameterized by θ. A mini-batch $\mathcal{B}$ consists of B labeled samples $\mathcal{X} = \{(x_b, y_b)|(x_b, y_b) \in D_L\}_{b=1}^{B}$ and μB unlabeled samples

$\mathcal{U} = \{u_b | u_b \in D_U\}_{b=1}^{\mu B}$, where μ is the ratio of unlabeled samples to the labeled samples in a mini-batch. The total loss $\mathcal{L}$ is minimized at each training iteration

$$\mathcal{L}(\mathcal{B}) = \mathcal{L}_L(\mathcal{X}) + \lambda_u \mathcal{L}_U(\mathcal{U}), \tag{1}$$

where $\mathcal{L}_L$ is a supervised loss, $\mathcal{L}_U$ is an unsupervised loss, and λ_u is an unlabeled loss ratio. Cross entropy is used for a supervised loss, and the type of $\mathcal{L}_U$ determines how to leverage the unlabeled samples. For example, entropy regularization [12] and pseudo-labeling enforce the predictions on unlabeled samples to have a low entropy, so that the decision boundary is located in the low-density area [25].

For an unlabeled sample $u \in \mathcal{U}$, the label prediction $\hat{p}(y|u) = \text{softmax}[W^\top h_\theta(u)]$ is given by the model with θ comprising K-class weight matrix $W = [w_1, w_2, ..., w_K] \in \mathbb{R}^{H \times K}$, where $h_\theta(u) \in \mathbb{R}^H$ denotes the penultimate features. We define a stochastic function of strong augmentation as α, and the set of strongly augmented samples for an unlabeled mini-batch as $\mathcal{A}_m(\mathcal{U}) = \{u'_i | u \in \mathcal{U}, u'_i = \alpha(u), 1 \leq i \leq m\}$, where m is the number of augmented view per an unlabeled sample in the mini-batch. Then, the consistency regularization, $\mathcal{R}_{CS}$, is defined as

$$\mathcal{R}_{CS}(\mathcal{U}) = \frac{1}{|\mathcal{A}_m(\mathcal{U})|} \sum_{u' \in \mathcal{A}_m(\mathcal{U})} \mathbb{1}[\max q_u > \delta] H(\hat{q}_u, \hat{p}(y|u')), \tag{2}$$

where u' is the augmented sample of $u \in \mathcal{U}$, δ is a confidence threshold, and H is the cross entropy loss. In the remaining parts of this paper, a strongly augmented sample of u is represented as u' for brevity. $\hat{q}_u$ is the pseudo-label of u' and defined as $\hat{q}_u = \arg\max q_u$, where $q_u = \text{sg}[\hat{p}(y|u)]$ and sg is the stop gradient. Note that the pseudo-label of $u' = \alpha(u)$ is determined by the label prediction on the sample without strong augmentation, u, for the reliability of pseudo-labeling.

The performance of consistency regularization highly depends on the choices of α and δ. Data augmentation encourages DNNs to learn the generalized representations with the local geometry of the data-manifold, assuming that the learned manifolds of different classes are well-separated [11, 30]. Therefore, the features having different pseudo-labels become well-separated, propagating the confident (pseudo-)labeling information into their neighbors on the data manifold. Determining the threshold δ is an inherent trade-off for the SSL performance, because the δ controls the balance of the reliability and the number of unlabeled samples leveraged. A higher value of δ is commonly used to avoid a confirmation bias, but it restricts unlabeled samples to be included in SSL training and can preclude the model from learning the transformation-invariant representations on the excluded samples [1]. It minimizes the entropy of only a sample using a confident pseudo-labeling for SSL training.

3.2. Training Inefficiency of Consistency Regularization

As the consistency regularization achieves high SSL performance competitive with the fully supervised setting, it requires a large number of training iterations even on small-scale datasets. For example, FixMatch [28] requires over 10,000 epochs to train WRN-28-2 [34] on the CIFAR-10 dataset. However, when the labels are fully provided, about 100 epochs are enough to learn the dataset under supervision.

Here, we analyze the consistency regularization to show its training inefficiency. Assume that $\hat{Q}_i$ is a set of strongly augmented samples assigned to the i-th class by the pseudo-label, $\hat{Q}_i = \{u' | u' \in \mathcal{A}_m(\mathcal{U}), u \in \mathcal{U}, \hat{q}_u = i\}$. The minus gradients of $\mathcal{R}_{CS}$ with respect to the features h_θ and to the i-th class weight vector w_i are as follows:

$$-\frac{\partial \mathcal{R}_{CS}}{\partial w_i} = \frac{1}{|\mathcal{A}_m(\mathcal{U})|} \sum_{u' \in \hat{Q}_i} \mathbb{1}[\max q_u > \delta] h_\theta(u')(1 - \hat{p}(i|u')), \tag{3}$$

$$-\frac{\partial \mathcal{R}_{CS}}{\partial h_\theta} = \frac{1}{|\mathcal{A}_m(\mathcal{U})|} \sum_{u' \in \mathcal{A}_m(\mathcal{U})} \mathbb{1}[\max q_u > \delta] \{ \sum_{i \neq \hat{q}_u} w_i \hat{p}(i|u') + w_{\hat{q}_u}(1 - \hat{p}(\hat{q}_u|u')) \}. \tag{4}$$

By this gradient analysis, we postulate that the inefficiency of the consistency regularization results from the exclusion of samples with unconfident pseudo-labels and the training bias on the confident pseudo-labels by the masking $\mathbb{1}[\max q_u > \delta]$. Figure 1(a) contains the interpretation of the gradient analysis. Here, we assume that the features with unconfident pseudo-labels are close to the decision boundary, considering the linearity of softmax classifier [5]. The class weight vector w_i is updated to the weighted sum of only confident features in Eq. (3). Then, the confident features in Eq. (4) are updated by the class weight vectors, so the features only having confident pseudo-labels become close together. However, the unconfident samples are excluded in the gradients computations, and the labeling information of confident samples cannot be effectively propagated into the unlabeled samples. In addition, the class weight vector is slowly changed due to the exclusion of unconfident samples, because the gradient in Eq. 3 is bounded by the confidence threshold, $-\frac{\partial \mathcal{R}_{CS}}{\partial w_i} < \frac{1}{|\mathcal{A}_m(\mathcal{U})|} \sum_{u' \in \hat{Q}_i} \mathbb{1}[\max q_{u_b} > \delta] h_\theta(u')(1 - \delta)$, where δ is typically selected as a high value such as 0.95. Thus, the model cannot leverage lots of unlabeled samples over the SSL training and requires a large number of training iterations to gradually increase the number of confident samples.

3.3. Contrastive Regularization for SSL

We propose *contrastive regularization* in Figure 1(b) to effectively leverage unlabeled samples for SSL. Even though Figure 1(b) describes the two-class classification,

the concept can be generalized to the setting of multiple classes, and all experiments in this study are also conducted on multi-class tasks. Different from the consistency regularization, the class clusters are formed by the features with both confident and unconfident pseudo-labels. Then, our method regularizes the hidden features of confident unlabeled samples to be moved toward the samples with unconfident pseudo-labels in the same cluster, and propagates the labeling information. At the same time, to leverage the unconfident samples without decreasing the confident threshold δ, the features having confident pseudo-labels pull the features of unconfident samples in the same cluster, while pushing the features in different clusters. It can achieve the entropy minimization for SSL, and unlabeled samples are beneficial with a small overlap of classes, since the contrastive regularization can learn well-clustered features that reduce the overlaps.

For this, we modify SupContrast [16], which is used for a supervised pretraining on large-scale labeled data, into SSL setting by adding a projection head after the penultimate features. We define the set of *pseudo*-positive pairs of u' as $\hat{P}(u') = \{p' | p' \in \mathcal{A}_m(\mathcal{U})/u', \hat{q}_p = \hat{q}_u\}$, where $\hat{q}_p$ and $\hat{q}_u$ are the pseudo-label of p' and u', respectively. Note that a pseudo-label of a strongly augmented sample u' is defined by the label prediction on the unlabeled sample u before strong augmentation. The positive sample pairs represent the samples whose pseudo-labels are the same, and the augmented samples in $\hat{P}(u')$ have the same pseudo-label with u'. Then, the contrastive regularization, $\mathcal{R}_{CR}$, is defined as follows:

$$\mathcal{R}_{CR}(\mathcal{U}) = \frac{1}{|\mathcal{A}_m(\mathcal{U})|} \sum_{u' \in \mathcal{A}_m(\mathcal{U})} \mathbb{1}[\max q_u > \delta'] r(u'), \quad (5)$$

$$r(u') = \frac{-1}{|\hat{P}(u')|} \sum_{p' \in \hat{P}(u')} \log \frac{\exp(\langle z_{u'}, z_{p'} \rangle / \tau)}{\sum_{v' \in \mathcal{A}_m(\mathcal{U})/u'} \exp(\langle z_{u'}, z_{v'} \rangle / \tau)}, \quad (6)$$

where δ' is a confidence threshold, τ is a temperature scaling parameter, and $z_{u'}$ is a *normalized* vector of the projection head. Our total loss is $\mathcal{L}(\mathcal{B}) = \mathcal{L}_L(\mathcal{X}) + \lambda_{CS}\mathcal{R}_{CS}(\mathcal{U}) + \lambda_{CR}\mathcal{R}_{CR}(\mathcal{U})$, where λ_{CS} and λ_{CR} is the loss ratio of consistency and contrastive regularization, respectively.

The features of confident samples move toward the centroid of its feature cluster, which consists of features having the same pseudo-labels, and pull the unconfident features in the same cluster by our contrastive regularization. Without the loss of generalizability, we notate the softmax score of $z_{p'}$ with $z_{u'}$ as $s[u', p']$, and assume the normalized vector $z = h$, and $\tau = 1$. For u' and $v' \in \mathcal{A}_m(\mathcal{U})/u'$, the minus gradients of $r(u')$ with respect to h_θ are as follows:

$$-\frac{\partial r(u')}{\partial h_\theta(u')} = \sum_{p' \in \hat{P}(u')} \left(\frac{1}{|\hat{P}(u')|} - s[u', p']\right) h_\theta(p') + R(u'), \quad (7)$$

$$-\frac{\partial r(u')}{\partial h_\theta(v')} = \begin{cases} \left(\frac{1}{|\hat{P}(u')|} - s[u', v']\right) h_\theta(u'), & \text{if } v' \in \hat{P}(u') \\ -s[u', v'] h_\theta(u'), & \text{if } v' \notin \hat{P}(u') \end{cases}, \quad (8)$$

where $R(u')$ is a remainder term and small enough. We attach the detailed derivation of Eq. (7) and (8) in Appendix A. If the u' has a confident pseudo-label as Eq. (7), the contrastive regularization updates its feature vector $h_\theta(u')$ toward the weighted sum of positive features both with confident and unconfident pseudo-labels. Different from the consistency regularization, the feature update of confident samples also considers the features with unconfident pseudo-labels in the same cluster. At the same time, in Eq. (8), the confident features pull the features of both confident and unconfident samples in the same cluster $\hat{P}(u')$, while pushing the features in different clusters. Although our contrastive regularization of a confident feature u' learns to aggregate positive samples with $s[u', v'] = 1/|\hat{P}(u')|$, the u' can push a positive sample v' of u' with a negative value in Eq. (8) *during training*, because all positive samples in a mini-batch are included in the denominator of the long term in Eq. (6). However, note that other negative samples in the different clusters still push v', avoiding a wrong cluster assignment by the negative values of Eq. (8) during training. Thus, the model can propagate the confident labeling information into the unlabeled samples, while learning well-clustered features for SSL [6].

Although our contrastive regularization utilizes the information of unconfident pseudo-labeling, the confirmation bias does not more increase than previous consistency regularization methods. According to Appendix C, the performance degradation by the memorization of wrong pseudo-labels occurs in the later stage of SSL training. In the early stage of training, our method learns well-clustered representations of unlabeled samples to effectively propagate labeling information of labeled samples and unlabeled samples with confident pseudo-labeling. Thus, the contrastive regularization can improve the SSL performance before the SSL model starts to memorize wrong pseudo-labels [2]. In addition, different from the consistency regularization, our method is performed on features of unlabeled samples, not directly on class predictions, to avoid the memorization of wrong labels by the contrastive regularization.

4. Experiments

We empirically validate that the contrastive regularization consistently improves the performance on standard SSL benchmarks such as SVHN, CIFAR-10, CIFAR-100, STL-10, and ImageNet with limited labels. We also show that the contrastive regularization is also robust to the open-set SSL setting, and an extensive ablation study is conducted in this section. For experiments, we use an exponential moving average (EMA) of model parameters [29] with 0.999 momentum and cosine learning rate scheduling

Table 1. Test accuracies (%) for SVHN and CIFAR-10 on five different runs with randomly selected labeled samples. The Asterisks mean that the results are from the previous studies [17, 20, 28].

	SVHN				CIFAR-10			
Method	20 labels	40 labels	250 labels	1000 labels	20 labels	40 labels	250 labels	4000 labels
MixMatch*	-	57.45±14.53	96.02±0.23	96.50±0.28	-	52.46±11.50	88.95±0.86	93.58±0.10
UDA*	-	43.75±20.51	94.31±2.76	97.54±0.24	-	70.95±5.93	91.18±1.08	95.12±0.18
ReMixMatch*	-	**96.66±0.20**	97.08±0.48	97.35±0.08	-	81.90±9.64	94.46±0.05	95.28±0.13
CoMatch*	-	-	-	-	81.85±5.56	91.51±2.15	-	-
FixMatch	90.05±8.01	94.83±2.24	97.28±0.66	97.46±0.09	74.98±11.38	91.24±3.72	94.67±0.28	95.57±0.05
FixMatch+CR	**94.96±4.77**	96.33±1.84	**97.55±0.08**	**97.61**±0.06	**88.26**±1.38	**94.31**±0.90	**94.96**±0.30	**95.84**±0.13
SelfMatch*	-	96.58±1.02	97.37±0.43	97.49±0.07	-	93.19±1.08	95.13±0.26	95.94±0.08
FixMatch+CR++	**96.88**±0.60	**97.05**±0.28	**97.95**±0.09	**98.11**±0.05	**94.24**±3.48	**95.26**±0.70	**96.00**±0.31	**96.68**±0.18

in [28] for all experiments. The training epochs are computed based on the batch size of unlabeled samples. For a fair comparison, we follow the experimental setting in the previous study [28], and attach the implementation details in Appendix B.

4.1. Classification of SVHN, CIFAR-10, CIFAR-100

To analyze the effect of contrastive regularization (CR), we reproduce FixMatch and UDA using Pytorch 1.6.0 [23]. For a fair comparison with previous studies, we use the encoder of WRN-28-2 (1.5M parameters) for SVHN and CIFAR-10, and a WRN-28-8 (23.4M parameters) for CIFAR-100. For SVHN and CIFAR-10, we also use WRN-28-8 (FixMatch+CR++) for comparison with SelfMatch [17] which uses over 21M parameters.

For the projection embedding z, we add a 2-layer MLP after the feature extractor h_θ, and its dimension sizes are 64 for WRN-28-2 and 256 for WRN-28-8. We use RandAugment [10] for strong data augmentation, and set $\lambda_{CS} = 1.0$. We use $\lambda_{CR} = 1.0$ for SVHN and CIFAR-10, and $\lambda_{CR} = 10.0$ for CIFAR-100. Following [28], FixMatch and UDA use 10,500 epochs of unlabeled data. FixMatch+CR uses 6,500 epochs for CIFAR-10 and SVHN, and 2,500 epochs for CIFAR-100 to achieve state-of-the-art results. Nevertheless, note that much less time needs to outperform FixMatch in the next section.

For SVHN and CIFAR-10, Table 1 shows that our method consistently improves the SSL performance of FixMatch on the same codebase. Consequently, the proposed FixMatch+CR achieves the state-of-the-art performance of WRN-28-2 except SVHN with 40 labels. Although FixMatch+CR cannot outperform the reported result of ReMixMatch [3] on SVHN with 40 labels, our method improves the test accuracy of FixMatch by 1.50%.

We emphasize that the contrastive regularization has remarkable performance gains. For the setting of 20 labels (2 labels per class), FixMatch+CR significantly improves the accuracy and the robustness to the selection of labeled samples. FixMatch+CR outperforms CoMatch [20], which has firstly reported the results on CIFAR-10 with 20 and 40 labels. In addition, WRN-28-2 with FixMatch+CR are competitive with SelfMatch, although the number of parameters is about 15 times smaller. When we increase the number of parameters into 23.4M, FixMatch+CR++ outperforms SelfMatch in all experimental settings of SVHN and CIFAR-10.

Table 2. Test accuracy (%) of WRN-28-8 on the CIFAR-100 dataset with 400, 2500, and 10000 labels.

	CIFAR-100		
Medthod	400 labels	2500 labels	10000 labels
UDA	48.02±2.66	70.50±0.53	77.07±0.33
UDA+CR	**49.91**±0.79	**72.12**±0.28	**78.58**±0.11
FixMatch	48.48±0.55	71.53±0.29	78.03±0.26
FixMatch+CR	**50.77**±0.79	**72.42**±0.37	**78.97**±0.23

Our method is also effective on the CIFAR-100 dataset with 400, 2,500, and 10,000 labels (Table 2). When the contrastive regularization is used along with UDA and FixMatch, it improves the performance and outperforms the previous methods. Note that the performance gains are significant and consistent regardless of the number of labels.

4.2. Classification of STL-10 and ImageNet

We evaluate our contrastive regularization on a larger scale of datasets such as STL-10 and ImageNet. We set $\lambda_{CS} = 1.0$ for SVHN and $\lambda_{CS} = 10.0$ for ImageNet, and $\lambda_{CR} = 10$ for the two. The STL-10 dataset includes 5,000 labeled and 100,000 unlabeled 96×96 images in 10 classes. We train WRN-37-2 (5.9M parameters) on STL-10 with five folds of 1,000 and 5,000 labels. 10,500 and 5,000 epochs are used for FixMatch and FixMatch+CR, respectively. The projection head uses 2-layer MLP with 256 dimensions. For 1,000 labels, FixMatch+CR improves the results of FixMatch in Table 3. For 5,000 labels, FixMatch+CR achieves 95.40%, improving 95.18% of FixMatch.

We also evaluate our method on the ImageNet dataset that includes about 1.3M training images in 1,000 object classes. We use a self-supervised and pretrained ResNet-

Table 3. Test accuracy (%) on the STL-10 and ImageNet datasets. Top-1 (top-5) accuracies are reported for ImageNet

	STL-10	ImageNet	
Method	1,000 labels	1% labels	10% labels
FixMatch	89.34±1.79	51.29 (72.48)	72.18 (89.98)
FixMatch+CR	**93.04**±0.42	**57.77 (78.12)**	**72.77 (90.15)**

(a) (b)

Figure 2. Results of FixMatch and FixMatch+CR with WRN-28-4 trained on the CIFAR-100 with 10000 labels. (a) Test accuracy over training time, (b) Silhouette score of penultimate features based on pseudo-labels.

50 model by MoCo v2 [9, 13], since reproducing FixMatch on ImageNet from scratch requires expensive cost such as about three days using 32 cores of TPU. We use 1,024 labeled and 5,120 unlabeled images in each mini-batch, and train a model in 300 epochs of unlabeled data. 2-layer MLP with 512 dimensions is used for the projection embedding. For 1% and 10% of labels, our contrastive regularization improves the accuracy of FixMatch in Table 3, and our method significantly improves the performance on the fewer labels. Thus, we conclude that our contrastive regularization is also effective on large-scale datasets.

4.3. Cost Efficiency of Contrastive Regularization

The contrastive regularization not only improves the accuracy, but also enhances the training efficiency of SSL. For a fair comparison of training time, four NVIDIA V100 GPUs are used to train both FixMatch and FixMatch+CR. In Figure 2(a), the accuracy of FixMatch gradually increases over the entire training time of 10,500 epochs. Although one iteration of FixMatch+CR takes about 1.5× more time than FixMatch due to the use of two strongly augmented views, FixMatch+CR only takes 31% of the total training time of FixMatch to achieve the best performance. Also, 7% of the training time of FixMatch is enough for FixMatch+CR to achieve the best performance of FixMatch (dashed line). For other datasets, 2,500 epochs for SVHN and CIFAR-10, 1,000 epochs for CIFAR-100, and 1,500 epochs for STL-10 are enough to outperform FixMatch of 10,500 epochs, as shown in Appendix C. Consequently, our method can save the training cost, reducing the training time and iterations.

We conjecture that the improved efficiency comes from the well-clustered representations by the contrastive regularization in the early stage of training. Figure 2(b) shows how features are well-clustered according to their pseudo-labels in terms of Silhouette score [24]. If the decision boundary lies in the low-density regions and the features are well-clustered, the score is closed to +1, otherwise it is closed to -1. For the features of strongly augmented samples, the clustering scores of FixMatch are near zero and it increases slowly after 40K iterations. However, the clustering score of FixMatch+CR increases fast in the early stage of training. In addition, the scores are much higher than those of FixMatch during the entire training. This means that our contrastive regularization is effective in feature clustering, especially in the early training stage, and eventually improves both the training efficiency and final performance.

Table 4. Test accuracy (%) with different sizes of the widen factor on the same random seed. 28 layers of WRN is trained on CIFAR-100 with 2500 labels.

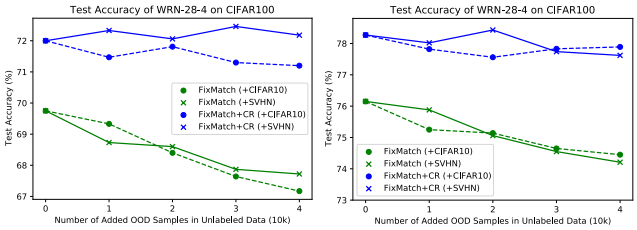
Widen Factor	1	2	4	8
# of Params	0.38M	1.48M	5.87M	23.40M
FixMatch	55.86	64.74	69.75	72.02
FixMatch+CR	**59.94**	**69.03**	**72.00**	**72.83**

Table 4 shows that the contrastive regularization is especially effective to a smaller model for SSL. When the contrastive regularization is applied to WRN-28-4 and WRN-28-2 with 2500 labels of the CIFAR-100, the accuracies are improved by 2.25% and 4.29%, respectively. Thus, the obtained accuracies of WRN-28-2 and WRN-28-4 with the contrastive regularization are comparable with those of WRN-28-4 and WRN-28-8 without it, respectively. Note that increasing the widen factor by two times leads to a four times larger number of trainable parameters.

4.4. Open-Set Semi-Supervised Learning

For a realistic evaluation, open-set SSL [22, 33] assumes that an unlabeled dataset includes out-of-distribution (OOD) samples, which are totally different from the training and test samples. Considering SVHN and CIFAR-10 as OOD of CIFAR-100, we add the OOD samples to the unlabeled data of CIFAR-100, and train WRN-28-4 on CIFAR-100 with 2500 and 10000 labels. Then, we evaluate the accuracy on the test data of CIFAR-100 according to the number of added OOD samples such as 10K, 20K, 30K, and 40K.

As shown in Figure 3, the contrastive regularization enhances the robustness of SSL to the OOD samples. FixMatch has severe degradation of accuracy as OOD samples are added into unlabeled data. However, FixMatch+CR avoids the accuracy degradation and always outperforms FixMatch regardless of the number of OOD samples, be-



(a) (b)

Figure 3. Open-set SSL results of WRN-28-4 on CIFAR-100 with (a) 2500 and (b) 10000 labels. OOD samples (SVHN, CIFAR-10) are added into the unlabeld samples.

cause FixMatch+CR effectively leverages unlabeled samples from in-distribution, when the number of labels is limited. (Table 1). Note that FixMatch+CR is more robust to the OOD samples from SVHN than those from CIFAR-10, since SVHN has a totally different class distribution from CIFAR-100 and less affects the discrimination of the CIFAR-100 classes.

4.5. Ablation Study

Effects of Hyper-parameter Settings. We conduct an extensive ablation study to understand the effects of the different components in our method. In Figure 4(a), the contrastive regularization improves the accuracy when the weight of the contrastive loss λ_{CR} is large enough (λ_{CS} is fixed to 1.0). Although an excessive large value of λ_{CR} deteriorates the test accuracy by increased confirmation bias, the performance is robust to the selection of the λ_{CR}.

Figure 4(b) shows the effect of the consistency regularization on the SSL performance, where the weight of contrastive loss λ_{CR} is fixed to 10.0. The test accuracy decreases when the weight of the consistency regularization λ_{CS} increases, since the relative effect of our contrastive regularization decreases. When the λ_{CS} becomes smaller than 1.0, the test accuracy is competitive with $\lambda_{CS} = 1.0$ and shows the effectiveness of our method.

Although the consistency regularization can completely be replaced with our contrastive regularization, we use the two regularizations to consistently achieve the state-of-the-art performance on different settings of the number of labeled samples. In Table 6, we remove the consistency regularization and evaluate our contrastive regularization alone (CR-only, $\lambda_{CS} = 0$) on CIFAR-10 with WRN-28-2. When 4000 labels are available, CR-only still outperforms FixMatch, but the performance of CR-only is degraded when 40 and 250 labels are used. Without the consistency regularization, the task-specific classification head is trained only by a supervised loss on labeled data $\mathcal{L}_L(\mathcal{X})$ in Eq. (1), and it can be easily overfitted when the number of labeled samples is few. Thus, our method is complementary to the consistency regularization to maximize the SSL performance.

Table 5. Test accuracies (%) on the numbers of views (m) per sample in $\mathcal{A}_m$ (CIFAR-100 with 10000 labels).

# of Views	$m = 1$	$m = 2$	$m = 3$
FixMatch	76.01	76.03	76.67
FixMatch+CR	76.08	**78.27**	77.83

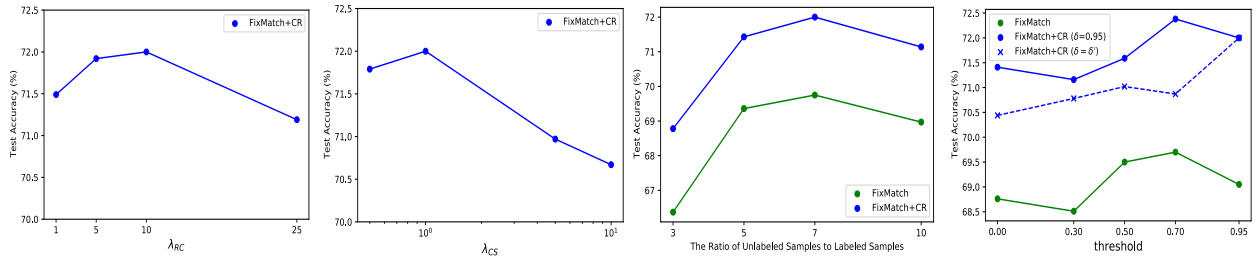
Table 6. Test accuracies (%) of contrastive regularization without consistency regularization on CIFAR-10

CIFAR-10	40 labels	250 labels	4000 labels
FixMatch	94.81	95.11	95.6
CR	91.51	94.41	95.89
FixMatch+CR	**95.32**	**95.39**	**95.92**

In Figure 4(c), the accuracies of both FixMatch and FixMatch+CR decrease when the ratio of unlabeled samples is low. This observation is consistent with the findings in UDA and FixMatch. It indicates that both consistency and contrastive approaches require a sufficiently large number of unlabeled samples in a mini-batch for high SSL performance.

Figure 4(d) shows that the confident threshold is related to the trade-off between the reliability of pseudo-labeling and the number of unlabeled samples leveraged. The confidence thresholds of the consistency and contrastive regularizations are denoted as δ in Eq. (2) and δ' in Eq. (5), respectively. The low value of δ worsens the performance of both FixMatch and FixMatch+CR because of the low reliability of pseudo-labeling. Although the test accuracy of FixMatch with $\delta = 0$ drops to 68.74%, only half of the training epochs are needed to achieve the performance, since it leverages all unlabeled samples regardless of the pseudo-labeling quality. When δ' becomes low, FixMatch+CR suffers from the confirmation bias to some degree, but the test accuracy of FixMatch+CR with $\delta = \delta' = 0$ still outperforms FixMatch, since our contrastive regularization effectively leverages unlabeled samples to improve the SSL performance. When δ keeps high value (0.95) and δ' becomes low, the performance of FixMatch+CR decreases, but the results with $\delta' = 0$ still significantly outperforms FixMatch with $\delta = 0.95$. The results imply that our contrastive regularization at the feature-level does not explicitly update the weights of the classifier, and therefore it shows robust results to the noisy pseudo-labels. However, as generating reliable pseudo-labels is still important to improve the SSL performance, we apply the selection mask with $\delta' > 0$ in our method. Note that, different from the consistency regularization, our contrastive regularization in Eq. (8) can still leverage unlabeled samples with both confident and unconfident pseudo-labels, while keeping high reliability of pseudo-labeling by the confidence threshold.

Effect of the Number of Views. FixMatch and Fix-



(a) λ_{CR} (b) λ_{CS} (c) μ (d) δ and δ'

Figure 4. The ablation study on the hyper-parameters (WRN-28-4, CIFAR-100 with 2500 labels): (a) contrastive regularization loss ratio, (b) consistency regularization loss ratio, (c) unlabeled sample ratio in a mini-batch, and (d) confidence thresholds.

Table 7. SSL performance of unsupervised contrastive learning as the SSL regularization.

	FixMatch	+NT-Xent	+CR
400 labels	49.42	32.27	**50.15**
2500 labels	69.75	54.44	**72.00**
10000 labels	76.15	67.02	**78.27**

Match+CR are compared with different numbers of augmented views m of $\mathcal{A}_m$. Note that at least two views of each sample are required for FixMatch+CR to assure the existence of a positive sample in each mini-batch [7]. Table 5 shows that the performance gain does not come from the solely increased number of views in our contrastive regularization. The accuracy of FixMatch is not improved by two augmented views and does not outperform FixMatch+CR although three augmented views are used. The results imply that the performance gain by our method does not depend on the increased number of views, but is from effectively leveraging more unlabeled data in SSL training shown in Figure 2.

Comparison with Unsupervised Contrastive Loss. We compare our method, which uses a contrastive loss with pseudo-labels for regularization, with the unsupervised contrastive loss (NT-Xent [7]). As an auxiliary task, a self-supervised pretext task is known to improve the performance of semi-supervised learning. Thus, we assume that unsupervised NT-Xent also improves the performance of semi-supervised learning, if the improvement by the contrastive regularization depends only on an auxiliary representation learning. Table 7 shows that the NT-Xent highly decreases the SSL performance. This might be due to that the task-agnostic instance discrimination tends to push away semantically similar instances in the same class.

5. Conclusion

Semi-supervised learning is important to learn a task with limited labels, while effectively leveraging unlabeled data in a task-specific way. In this work, we show that the SSL training of the previous consistency regularization is biased on unlabeled samples only with confident pseudo-labeling by a selection mask. Thus, we propose contrastive regularization that significantly improves SSL performance and can be used along with the consistency regularization by minimal change of implementation. On various SSL benchmarks, the contrastive regularization not only improves the test accuracy but also significantly reduces the number of training iterations to achieve high performance. Especially, our method shows more effective results on a dataset containing fewer labels or out-of-distribution samples.

In future work, our approach can be applied to other SSL methods based on pseudo-labels [8, 32] on large-scale datasets. However, our method still has a limitation on the memory efficiency for large-scale datasets due to large batch size, although the contrastive regularization improves training efficiency and the SSL performance. Thus, it is also worth exploration for leveraging large-scale unlabeled datasets in a task-specific way.

6. Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No.2018-0-01398: Development of a Conversational, Self-tuning DBMS; No.2021-0-00537: Visual Common Sense).

References

[1] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020. 1, 3

[2] Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, pages 233–242. PMLR, 2017. 4

[3] 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 *International Conference on Learning Representations*, 2019. 1, 2, 5

[4] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[5] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006. 3

[6] Vittorio Castelli and Thomas M Cover. The relative value of labeled and unlabeled samples in pattern recognition with an unknown mixing parameter. *IEEE Transactions on information theory*, 42(6):2102–2117, 1996. 4

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 8

[8] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems*, 33:22243–22255, 2020. 1, 2, 8

[9] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 6

[10] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020. 2, 5

[11] Atin Ghosh and Alexandre H Thiery. On data-augmentation and consistency-based semi-supervised learning. In *International Conference on Learning Representations*, 2021. 1, 3

[12] Yves Grandvalet, Yoshua Bengio, et al. Semi-supervised learning by entropy minimization. In *CAP*, pages 281–296, 2005. 3

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 2, 6

[14] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label propagation for deep semi-supervised learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5070–5079, 2019. 2

[15] Shayan Jawed, Josif Grabocka, and Lars Schmidt-Thieme. Self-supervised learning for semi-supervised time series classification. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 499–511. Springer, 2020. 2

[16] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33, 2020. 4

[17] 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 arXiv:2101.06480*, 2021. 5

[18] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. Big transfer (bit): General visual representation learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pages 491–507. Springer, 2020. 2

[19] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, 2013. 1, 2

[20] Junnan Li, Caiming Xiong, and Steven Hoi. Comatch: Semi-supervised learning with contrastive graph regularization. *arXiv preprint arXiv:2011.11183*, 2020. 1, 2, 5

[21] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1979–1993, 2018. 2

[22] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *Advances in Neural Information Processing Systems*, 31:3235–3246, 2018. 2, 6

[23] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 5

[24] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987. 6

[25] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *Advances in neural information processing systems*, 29:1163–1171, 2016. 3

[26] Laine Samuli and Aila Timo. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017. 2

[27] Weiwei Shi, Yihong Gong, Chris Ding, Zhiheng MaXiaoyu Tao, and Nanning Zheng. Transductive semi-supervised deep learning using min-max features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 299–315, 2018. 2

[28] 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. *Advances in Neural Information Processing Systems*, 33, 2020. 1, 2, 3, 5

[29] Antti Tarvainen and Harri 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. 4

[30] Vikas Verma, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, pages 3635–3641. AAAI Press, 2019. 3

[31] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, Pierre-Antoine Manzagol, and Léon Bottou. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(12), 2010. 2

[32] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33, 2020. 2, 8

[33] Qing Yu, Daiki Ikami, Go Irie, and Kiyoharu Aizawa. Multi-task curriculum framework for open-set semi-supervised learning. In *European Conference on Computer Vision*, pages 438–454. Springer, 2020. 2, 6

[34] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference 2016*. British Machine Vision Association, 2016. 3

[35] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3712–3722, 2018. 2

[36] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1476–1485, 2019. 2

[37] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 2