

Semi-supervised Learning via Conditional Rotation Angle Estimation

Hai-Ming Xu, Lingqiao Liu, Dong Gong

The University of Adelaide, Australia

{hai-ming.xu;lingqiao.liu}@adelaide.edu.au;edgong01@gmail.com

Abstract

*Self-supervised learning (SIfSL), aiming at learning feature representations through ingeniously designed pretext tasks without human annotation, has achieved compelling progress in the past few years. Very recently, SIfSL has also been identified as a promising solution for semi-supervised learning (SemSL) since it offers a new paradigm to utilize unlabeled data. This work further explores this direction by proposing to couple SIfSL with SemSL. Our insight is that the prediction target in SemSL can be modeled as the latent factor in the predictor for the SIfSL target. Marginalizing over the latent factor naturally derives a new formulation which marries the prediction targets of these two learning processes. By implementing this idea through a simple-but-effective SIfSL approach – rotation angle prediction, we create a new SemSL approach called Conditional Rotation Angle Estimation (CRAE). Specifically, CRAE is featured by adopting a module which predicts the image rotation angle **conditioned on the candidate image class**. Through experimental evaluation, we show that CRAE achieves superior performance over the other existing ways of combining SIfSL and SemSL. To further boost CRAE, we propose two extensions to strengthen the coupling between SemSL target and SIfSL target in basic CRAE. We show that this leads to an improved CRAE method which can achieve the state-of-the-art SemSL performance.*

1. Introduction

The recent success of deep learning is largely attributed to the availability of a large amount of labeled data. However, acquiring high-quality labels can be very expensive and time-consuming. Thus methods that can leverage easily accessible unlabeled data become extremely attractive. Semi-supervised learning (SemSL) and self-supervised learning (SIfSL) are two learning paradigms that can effectively utilize massive unlabeled data to bring improvement to predictive models.

SemSL assumes that a small portion of training data is provided with annotations and the research question is how

to use the unlabeled training data to generate additional supervision signals for building a better predictive model. In the past few years, various SemSL approaches have been developed in the context of deep learning. The current state-of-the-art methods, e.g. MixMatch [2], unsupervised data augmentation [11], converge to the strategy of combining multiple SemSL techniques, e.g. II-Model [10], Mean Teacher [18], mixup [21], which have been proved successful in the past literature.

SIfSL aims for a more ambitious goal of learning representation without any human annotation. The key assumption in SIfSL is that a properly designed pretext predictive task which can be effortlessly derived from data itself can provide sufficient supervision to train a good feature representation. In the standard setting, the feature learning process is unaware of the downstream tasks, and it is expected that the learned feature can benefit various recognition tasks. SIfSL also offers a new possibility for SemSL since it suggests a new paradigm of using unlabeled data, i.e., use them for feature training. Recent work [20] has shown great potential in this direction.

This work further advances this direction by proposing to couple SIfSL with SemSL. The key idea is that the prediction target in SemSL can serve as a latent factor in the course of predicting the pretext target in a SIfSL approach. The connection between the predictive targets of those two learning processes can be established through marginalization over the latent factor, which also implies a new method of SemSL. Specifically, we implement this idea by extending the rotation angle estimation [5] – a recently proposed SIfSL approach for image recognition, and the key component of our method is a module that estimates the rotation angle conditioned on the class of the input image. Therefore, we call our method Conditional Rotation Angle Estimation (CRAE). To further promote the mutual dependency of the prediction task of SemSL and SIfSL, we propose two extensions: one sharpens the estimated posterior probability of image classes on unlabeled data and encourages the conditional rotation estimator being more specialized towards its conditioned class; another modifies the rotation estimation task to increase the difficulty of guessing the rotation

angle without using class information, and thus make overall loss minimization depend more on correct estimation of the image class.

Through experiments, we show that the proposed CRAE achieves significantly better performance than the other SIfSL-based SemSL approaches, and the extended CRAE is on par with the state-of-the-art SemSL methods. In summary, the main contributions of this paper are as follows:

- We propose a new SemSL idea by seamlessly coupling SIfSL and SemSL. Implementing this idea with a SIfSL approach, we create a new SemSL approach (CRAE) that can achieve superior performance than other SIfSL-based SemSL methods.
- We further make two extensions over the basic CRAE to boost its performance. The resulted new method achieves the state-of-the-art performance of SemSL.

2. Related Work

Our work is closely related to both SemSL and SIfSL.

SemSL is a long-standing research topic which aims to learn a predictor from a few labeled examples along with abundant of unlabeled ones. SemSL based on different principals are developed in the past decades, e.g., "transductive" models [4, 6], multi-view style approaches [3, 23] and generative model-based methods [7, 17], etc. Recently, the consistency regularization based methods have become quite influential due to their promising performance in the context of deep learning. Specifically, Π -Model [10] requires model's predictions to be invariant when various perturbations are added to the input data. Mean Teacher [18] enforces a student model producing similar output as a teacher model whose weights are calculated through the moving average over the weight of student model. Virtual Adversarial Training [12] encourages the predictions for input data and its adversarially perturbed version to be consistent. More recently, mixup [21, 19] has emerged as a powerful SemSL regularization method which requires the output of mixed data to be close to the output mixing of original images. In order to achieve good performance, most state-of-the-art approaches adopt the strategy of combining several existing techniques together. For example, Interpolation Consistency Training [19] incorporates Mean Teacher into the mixup regularization, MixMatch [2] adopts a technique that uses fused predictions as pseudo prediction target as well as the mixup regularization. Unsupervised data augmentation [11] upgrades Π -Model with advanced data augmentation methods.

SIfSL is another powerful paradigm which learns feature representations through training on pretext tasks whose labels are not human annotated [8]. Various pretext tasks are designed in different approaches. For example, image inpainting [15] trains model to reproduce an arbitrary masked

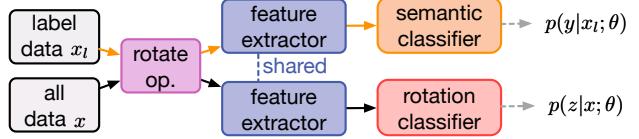


Figure 1. Structure of S^4L method.

region of the input image. Image colorization [22] encourages model to perform colorization of an input grayscale image. Rotation angle prediction [5] forces model to recognize the angle of a rotated input image. After training with the pretext task defined in a SIfSL method, the network is used as a pretrained model and can be fine-tuned for a downstream task on task-specific data. Generally speaking, it is still challenging for SIfSL method to achieve competitive performance to fully-supervised approaches. However, SIfSL provides many new insights into the use of unlabeled data and may have a profound impact to other learning paradigms, such as semi-supervised learning.

SIfSL based SemSL is an emerging approach which incorporates SIfSL into SemSL. The most straightforward approach is to first perform SIfSL on all available data and then fine-tune the learned model on labeled samples. S^4L [20] is a newly proposed method which jointly train the downstream task and pretext task in a multi-task fashion without breaking them into stages. In this paper, we further advance this direction through proposing a novel architecture which explicitly links these two tasks together and ensure that solving one task is beneficial to the other.

3. Our approach

3.1. Coupling SemSL with SIfSL

In SemSL, we are given a set of training samples $\{x_1, x_2, \dots, x_n\} \in X$ with only a few of them $X_l = \{x_1, x_2, \dots, x_l\} \in X$ annotated with labels $\{y_1, y_2, \dots, y_l\} \in Y_l$ (usually $l \ll n$ and y is considered as class label here). The goal of a SemSL algorithm is to learn a better posterior probability estimator over y from both labeled and unlabeled training samples, i.e., learning $p(y|x; \theta)$ with θ as model parameters, SIfSL aims to learn feature representations via a pretext task. The task usually defines a target z , which can be derived from the training data itself, e.g., rotation angle of the input image. Once z is defined, SIfSL is equivalent to training a predictor to model $p(z|x; \theta)$. There are two existing schemes to leverage SIfSL for SemSL. The first is to use SIfSL to learn the feature from the whole training set and then fine-tune the network on the labeled part. The other is jointly optimizing the tasks of predicting y and z , as in the recently proposed S^4L method, shown in Figure 1. S^4L constructs a network with two branches and a shared feature extractor. One branch for

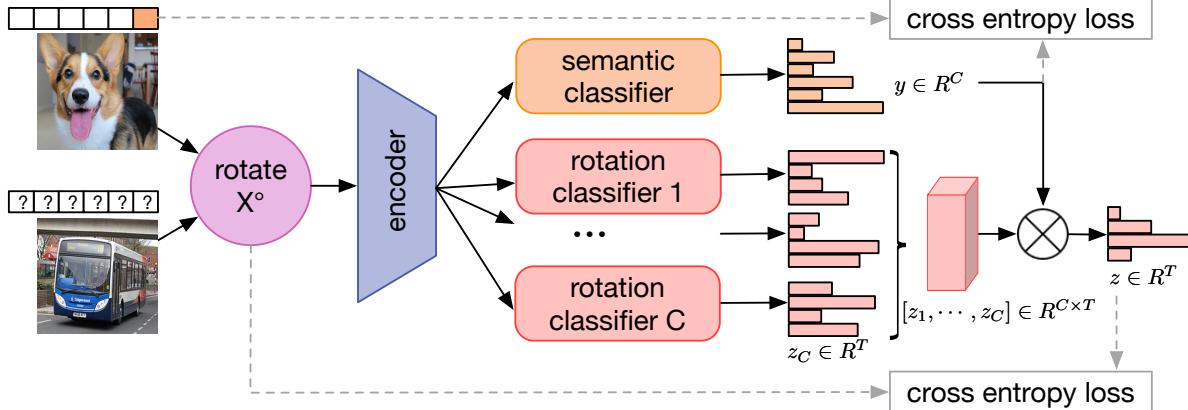


Figure 2. Illustration of our proposed CARE method. CRAE creates multiple conditional rotation estimation branches, one for each class. The semantic classifier acts as a switch to softly select (through weighted summation) the estimation of z from each rotation classifier.

modeling $p(y|x; \theta)$ and another for $p(z|x; \theta)$. However, in both methods the prediction $p(z|x; \theta)$ of the pretext target z is implicitly related to the task of predicting y .

In this work, we are interested in an explicit connection between the prediction target z and y . To establish this connection, we treat y as the latent factor in $p(z|x; \theta)$ and factorize $p(z|x; \theta)$ through marginalization:

$$p(z|x; \theta) = \sum_y p(z, y|x; \theta) = \sum_y p(z|x, y; \theta)p(y|x; \theta). \quad (1)$$

Eq. 1 suggests that the pretext target predictor $p(z|x; \theta)$ can be implemented as two parts: a model to estimate $p(y|x; \theta)$ and a model to estimate z conditioned on both x and y , i.e., $p(z|x, y; \theta)$. For the labeled samples, the ground-truth y is observed and can be used for training $p(y|x; \theta)$. For unlabeled samples, the estimation from $p(y|x; \theta)$ and $p(z|x, y; \theta)$ will be combined together to make the final prediction about z . Consequently, optimizing the loss for $p(z|x; \theta)$ will also provide gradient to back-propagate through $p(y|x; \theta)$. This is in contrast to the case of S^4L , where the gradient generated from the unlabeled data will not flow through $p(y|x; \theta)$.

3.2. CRAE: SemSL via conditional rotation angle estimation

In the following part, we describe our implementation of the above idea. Eq. 1 may apply to a range of SIfSL approaches. In this work, we employ rotation angle estimation [5]—a recently proposed SIfSL approach for image recognition. It randomly rotates the input image by one of the four possible rotation angles ($\{0^\circ, 90^\circ, 180^\circ, 270^\circ\}$) and requires the network to give a correct prediction of the rotation angle. Despite being extremely simple, this method works surprisingly well in practice. The underlying logic is that to correctly predict the rotation angle, the network

needs to recognize the canonical view of objects from each class and thus enforces the network to learn informative patterns of each image category.

There are two components in the Eq. 1: one is $p(y|x)$ and the other is $p(z|y, x)$ (hereafter, θ is omitted for simplicity). We implement both components as prediction heads attaching to the same feature extraction backbone. $p(y|x)$ is simply implemented as a classifier branch. For rotation angle estimation, the pretext target z is the rotation angle of the input image. $p(z|x, y)$ is thus a rotation angle estimator conditioned on a candidate image class y . In this work, we realize $p(z|x, y)$ by allocating a rotation angle estimation branch for each candidate class. Since this method is featured by the conditional rotation angle estimation (CRAE), we call our method CRAE. To simplify the notation, we denote each rotation angle estimation branch as $R_k(x) = p(z|x, y = k)$, $k = 1 \dots C$ or simply R_k . Follow Eq. 1, the prediction from each branch is then aggregated with the aid of $p(y|x)$ for the final prediction of z , that is, $p(z|x) = \sum_{k=1}^C R_k(x)p(y = k|x)$. A more detailed schematic illustration of the CRAE method is shown in Figure 2. Note that after training, all the rotation angle estimation branches are discarded. Thus our method will not introduce additional computational cost at the test stage.

Formally, the training objective of CRAE is given as follows:

$$L = \sum_{(x_l, y_l) \in \mathcal{L}} L_{ce}(p(y|x_l), y_l) + \eta \sum_{x \in \mathcal{U} \cup \mathcal{L}} L_{ce}(p(z|x), \hat{z}), \quad (2)$$

where \mathcal{L} and \mathcal{U} are labeled and unlabeled datasets respectively. \hat{z} is the ground truth of rotation angle. L_{ce} denotes the cross-entropy loss. Note that the estimation of $p(z|x)$ involves the term $p(y|x)$ as shown in Eq. 1. For labeled data, we use ground-truth y_l , i.e., the one-hot vector to replace the estimated $p(y|x_l)$.

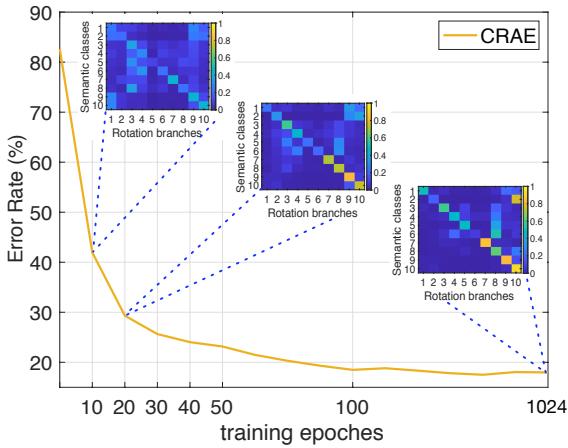


Figure 3. Visualization for error rates (%) of classifier and confusion matrix change during the training progress of CRAE on CIFAR-10 with 250 labels. If the confusion matrix is closer to a diagonal matrix, it indicates R_k is more specialized towards their corresponding class. To validate this, we plot the prediction error rates of the semantic classifier branch on unlabeled data and the confusion matrix in Figure 3, where the (i, j) entry of the confusion matrix indicates the chance that R_j gives the best estimation for z given the true class being i . It indicates R_k is more specialized to k if it is more close to a diagonal matrix. As seen, at the first few epochs, the error rates of $p(y|x)$ is high and R_k is less specialized.

3.3. Analyzing the training process of CRAE

In Eq. 1, the benefit of our method was explained as being able to generate gradient to update $p(y|x)$ on unlabeled data. But why this ability will lead to better estimation of $p(y|x)$? The underlying mechanism is actually more involved. In the following, we analyze the training process of CRAE with more details.

There are three key insights which are essential to understand CRAE: **(1) a good estimation of $p(y|x)$ will make each R_k more specialized to the k -th class.** In Eq. 1, $p(y|x)$ can be viewed as a switch to (softly) select the relevant R_k . When $p(y|x)$ is close to the ground-truth distribution – an one-hot vector (with the only “1” given to the correct class of x), only the R_k corresponding to the correct class will be used for predicting z and being updated. This will make R_k be updated solely (or more adequately in reality) for images from the k -th class, and consequently R_k tends to produce better estimation of z if the class label of x is k . **(2) on the other hand, a specialized R_k will push $p(y|x)$ to give higher probability to the true class of x , and thus improve the estimation of $p(y|x)$.** This is because in order to optimize the overall estimation of z , the training process tends to select R_k whose prediction is closest to the ground-truth rotation angle. Follow the conclusion in (1) – R_k tends to give better estimation of z if the true image class is k , the training process will push $p(y|x)$ to attain higher value for the true class. **(3) the update of R_k and the branch for estimating $p(y|x)$ will help learning better feature representations.** R_k is essentially a rotation angle estimator, according to [5], optimizing the

rotation angle estimation will induce a better feature representation. $p(y|x)$ is directly related to predicting target of interest. The update of $p(y|x)$ will simultaneously improve the classifier branch and its attached feature backbone. In some sense, updating feature representation through $p(y|x)$ is ideal since ultimately the feature will be used for predicting y . In summary, we can see that the update of $p(y|x)$ and R_k is mutually dependent and reinforce to each other.

In practice, the estimation of $p(y|x)$ is far from perfect at the beginning of training due to the limited number of labeled samples. At this stage, R_k is less specialized towards their corresponding class. To validate this, we plot the prediction error rates of the semantic classifier branch on unlabeled data and the confusion matrix in Figure 3, where the (i, j) entry of the confusion matrix indicates the chance that R_j gives the best estimation for z given the true class being i . It indicates R_k is more specialized to k if it is more close to a diagonal matrix. As seen, at the first few epochs, the error rates of $p(y|x)$ is high and R_k is less specialized.

However, since R_k also has the effect of learning features, the shared feature extractor will be improved through the course of training. With a better feature representation, the estimation of $p(y|x)$ will also be improved. The improved $p(y|x)$ branch will then follow the analysis above to make R_k specialized for k and this will in turn reinforce the estimation of $p(y|x)$. As seen from Figure 3, the error rates of $p(y|x)$ decreases and the confusion matrix becomes closer to diagonal matrix with more training epochs.

3.4. Extensions to CRAE

In the following, we propose two extensions to further improve the training process of CRAE from the perspectives of $p(y|x)$ and $p(z|x, y)$, respectively.

Extension 1: Encouraging a sharper $p(y|x)$

Based on analysis in Section 3.3, we can see that $p(y|x)$ acts as a (soft) selector for R_k . A sharper $p(y|x)$ estimation, that is, an estimation of $p(y|x)$ closer to an one hot vector, will encourage R_k to be updated mainly from its corresponding class, and thus make it more specialized to k . In this section, we create an additional loss to encourage $p(y|x)$ to be a sharper distribution. Specifically, we rotate each image in four angles within one batch (the predictions of the rotated image are the byproduct of CRAE) and obtain the average \bar{p} of the predicted distributions across these four rotated samples. Then we perform a sharpening operation over \bar{p} to create a sharpened target \hat{p} for $p(y|x)$:

$$\hat{p}_i = \frac{(\bar{p}_i)^{\frac{1}{T}}}{\sum_{j=1}^C (\bar{p}_j)^{\frac{1}{T}}}, \quad (3)$$

where C is the number of classes and $T \in (0, 1]$ is a temperature hyper-parameter. Then we use the cross entropy between \hat{p} and $p(y|x)$ as an additional loss to encourage a

sharper $p(y|x)$. Note that since the \hat{p} is obtained by averaging estimations from four augmented input images, it is expected to be more accurate than $p(y|x)$. Thus this loss also has the effect of encouraging a more reliable estimation of $p(y|x)$. Empirically, we find it works better than other sparse regularizers such as entropy minimization loss.

Extension 2: Making $p(z|x)$ depend more on $p(y|x)$

CRAE provides indirect supervision for $p(y|x)$ based on the condition that the correct estimation of $p(y|x)$ will improve the overall estimation of z . Thus if the optimization of the overall estimation $p(z|x)$ relies more on the correct estimation of $p(y|x)$, then CRAE is more likely to provide stronger supervision for $p(y|x)$.

To achieve this goal, we propose another extension by modifying the conditional rotation estimation task. Specifically, we require the rotation prediction branch to predict rotation angle for a mixed version of the rotated image, that is, we randomly mix the input image x_i with another randomly sampled rotated image x_j via $x_{\text{mix}} = \alpha x_i + (1 - \alpha)x_j$, with α sampled from $[0.5, 1]$. Meanwhile, the class prediction $p(y|x_i)$ is calculated from the unmixed version of the input x_i . In such a design, the network needs to recognize the rotation angle of the target object with the noisy distraction from another image. The purpose of introducing this modified task is to make the SlfSL task more challenging and more dependent on the correct prediction from $p(y|x)$.

To see this point, let's consider the following example. Letter 'A' is rotated with 270° and is mixed with letter 'B' with rotation 90° . Directly predicting the rotation angle for this mixed image encounters an ambiguity: whose rotation angle, A's or B's, is the right answer? In other words, the network cannot know which image class is the class-of-interest. This ambiguity can only be resolved from the output of $p(y|x)$ since its input is unmixed target image. Therefore, this improved rotation prediction task relies more on the correct prediction from the semantic classifier and training through CRAE is expected to give stronger supervision signal to $p(y|x)$. Note that although this scheme also uses mix operation, **it is completely different from mixup [21]**. The latter constructs a loss to require the output of the mixed image to be mixed version of the outputs of original images. This loss is not applied in our method.

In summary, the training objective for the extended CRAE is as follows

$$L = \sum_{(x_l, y_l) \in \mathcal{L}} L_{ce}(p(y|x_l), y_l) + \eta_1 \sum_{x \in \mathcal{U} \cup \mathcal{L}} L_{ce}(p(z|x_{\text{mix}}), \hat{z}) \\ + \eta_2 \sum_{x_u \in \mathcal{U}} L_{ce}(p(y|x_u), \hat{p}) \quad (4)$$

where x_{mix} indicates the mixed input image and \hat{p} in the last term is the sharpened target in Eq 3.

3.5. Other Implementation details

One potential obstacle of our model is that the quantity of parameters in the rotation branches would increase significantly with a large C . To tackle this, we propose to perform dimension reduction for the features feeding into the rotation predictors. Results in Figure 4 show that this scheme is effective as our performance will not drop even when the dimension is reduced from 2048 to 16. Also, we find for basic CRAE, it is beneficial to add an additional semantic classifier branch to the feature extractor, and learn it via the loss incurred from the labeled data only. At test time, this classifier will be directly used for testing. This treatment is similar to S^4L and can bring a slight performance gain for basic CRAE. We postulate the reason is that the indirect supervision generated for the original $p(y|x)$ branch from CRAE training can be noisy comparing with supervision directly generated from the ground-truth y . It is better to use such a branch just for feature learning since feature learning is more tolerant to noisy supervision. More implementation details can be found in the supplementary material.

4. Experiments

In this section, we conduct experiments to evaluate the proposed CRAE method. The purpose of our experiments is threefolds: (1) to validate if CRAE is better than other SlfSL-based SemSL algorithms. (2) to compare **CRAE** and extended CRAE (denoted as **CRAE+** hereafter) against the state-of-the-art SemSL methods. (3) to understand the contribution of various components in CRAE and CRAE+.

4.1. Experimental details

To make a fair comparison to recent works, different experimental protocols are adopted for different datasets. Specifically, for CIFAR-10 and CIFAR-100 [9] and SVHN [13], we directly follow the settings in [2]. For ILSVRC-2012 [16], our settings are identical to [20] except for data pre-processing operations for which we only use the inception crop augmentation and horizontal mirroring. We ensure that all the baselines are compared under the same setting. Followed the standard settings of SemSL, the performance with different amount of labeled samples are tested. For CIFAR-10 and SVHN, sample size of labeled images is ranged in five levels: $\{250, 500, 1000, 2000, 4000\}$. For CIFAR-100, 10000 labeled data is used for training. For ILSVRC-2012, 10% and 1% of images are labeled among the whole dataset. More experimental details can be found in the supplementary material.

4.2. Comparing with SlfSL-Based SemSL methods

Firstly, we compare CRAE to other SlfSL-based SemSL algorithms on five datasets: CIFAR-10, CIFAR-100, SVHN, SVHN+Extra and ILSVRC-2012.

Table 1. Error rates (%) for SlfSL based SemSL on CIFAR-10.

| # Labels | 250 | 500 | 1000 | 2000 | 4000 |
|--------------|--------------|--------------|--------------|--------------|-------------|
| Labeled-only | 56.76 | 47.24 | 36.09 | 29.90 | 19.65 |
| Fine-tune | 53.66 | 35.18 | 28.17 | 22.00 | 17.39 |
| S^4L | 32.66 | 28.23 | 22.55 | 18.91 | 15.71 |
| CRAE | 17.07 | 15.57 | 12.90 | 10.72 | 9.07 |

Table 2. Error rates (%) for SlfSL based SemSL on SVHN and SVHN+Extra.

| | # Labels | 250 | 500 | 1000 | 2000 | 4000 |
|--------|--------------|-------------|-------------|-------------|-------------|-------------|
| SVHN | Labeled-only | 23.27 | 17.85 | 12.84 | 9.65 | 7.26 |
| | Fine-tune | 16.21 | 11.54 | 8.31 | 7.30 | 5.61 |
| | S^4L | 10.48 | 8.32 | 7.21 | 6.15 | 5.74 |
| | CRAE | 9.23 | 7.61 | 5.76 | 4.93 | 4.74 |
| +Extra | Fine-tune | 13.11 | 9.52 | 7.74 | 6.17 | 5.57 |
| | S^4L | 8.47 | 6.58 | 5.44 | 4.30 | 3.87 |
| | CRAE | 8.24 | 5.42 | 4.61 | 4.13 | 3.79 |

Two SlfSL-based SemSL baseline approaches are considered: 1) **Fine-tune**: taking the model pre-trained on the pretext task as an initialization and fine-tuning with a set of labeled data. We term this method Fine-tune in the following sections. 2) S^4L : S^4L method proposed in [20]. Note that our extension 1 uses additional training loss other than the SlfSL loss on the labeled data, and extension 2 uses a modified SlfSL task. Those factors are not considered in other SlfSL-based SemSL baselines. To make a fair comparison, we only use our basic CRAE model in the comparison in this subsection. Also, as a reference, we report the performance obtained by only using the labeled part of the dataset for training, denoting as **Labeled-only**. The experimental results are as follows:

CIFAR-10 The results are presented in Table 1. We find that the “Fine-tune” strategy leads to a mix amount of improvement over the “Labeled-only” case. It is observed that a large improvement can be obtained when the amount of labeled samples is ranged from 500 to 2000 but not on 250 and 4000’s settings. It might be because on the one hand too few labeled samples are not sufficient to perform an effective fine-tuning while on the other hand the significant improvement diminishes after the sample size increase. In comparison, S^4L achieves much better accuracy for the case of using few samples. This is largely benefited from its down-stream-task awareness design. Our CRAE method achieves significantly better performance than those two ways of incorporating SlfSL for SemSL and always halves the test error of S^4L in most cases.

SVHN and SVHN+Extra Table 2 shows the results of each method. Apparently, both Fine-tune and S^4L methods can achieve much better performance than the Labeled-only baseline especially for case with few labeled samples, i.e.,

Table 3. Error rates (%) for SlfSL based SemSL on CIFAR-100.

| Methods | Labeled-only | Fine-tune | S^4L | CRAE |
|-----------|--------------|-----------|--------|--------------|
| CIFAR-100 | 46.53 | 34.11 | 33.89 | 30.68 |

Table 4. ILSVRC-2012 accuracies (%) for SlfSL based SemSL.

| # Labels | 10% | | 1% | |
|--------------|--------------|--------------|--------------|--------------|
| | Top1 | Top5 | Top1 | Top5 |
| Labeled-only | - | 80.43 | - | 48.43 |
| Fine-tune | - | 78.53 | - | 45.11 |
| S^4L | - | 83.82 | - | 53.37 |
| Labeled-only | 59.16 | 83.07 | 41.26 | 69.07 |
| S^4L | 63.84 | 86.28 | 46.90 | 74.16 |
| CRAE | 65.34 | 87.07 | 49.26 | 75.57 |

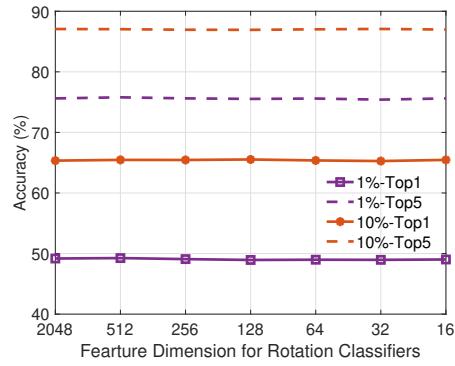


Figure 4. ILSVRC-2012 accuracy (%) changes with various dimension sizes of input feature channel for CRAE.

only 250 or 500 labeled data. S^4L outperforms Fine-tune method on most of the settings except for obtaining comparable results on 4000 labels setting. In comparison, the proposed CRAE still manages to produce superior accuracy than S^4L on all those settings. With more unlabeled samples from SVHN’s Extra dataset, all methods gain a further improvement. Continuously, our method slightly excels both Fine-tune and S^4L methods.

CIFAR-100 As shown in Table 3, it is obvious that all SlfSL-based SemSL methods can have better accuracy than that of Labeled-only and S^4L leads to a marginal improvement over Fine-tune. Again, the proposed CRAE method performs better than those baselines.

ILSVRC-2012 Table 4 presents the results of each method. The top block of Table 4 shows the reported results in the original S^4L paper [20] and we also re-implement S^4L based on the code of [8]. Due to the difference of data pre-processing, results in the upper block cannot be directly compared to those below. Again, we have observed that CRAE is consistently superior to S^4L in all settings.

As mentioned in Section 3.5, for saving the computa-

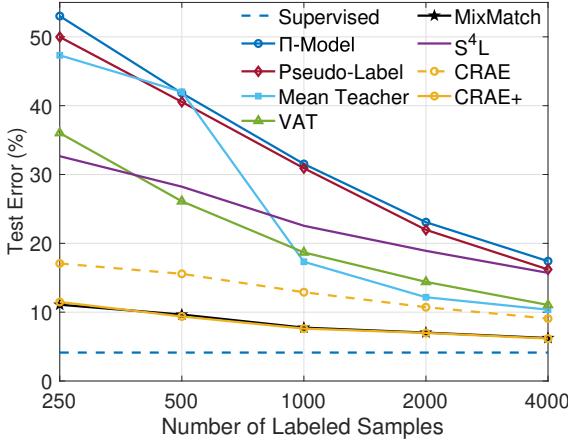


Figure 5. CIFAR-10 error rates (%) of CRAE+ and SOTA SemSL.

Table 5. Error rates (%) comparison of CRAE+ to SOTA SemSL methods on CIFAR-100.

| Methods | SWA | MixMatch | CRAE+ |
|-----------|------|----------|--------------|
| CIFAR-100 | 28.8 | 25.88 | 25.69 |

tional cost, we propose to reduce the dimensionality of features fed into the rotation angle predictors when there is a large number of classes. In Figure 4, we demonstrate the effect of this scheme. As seen, the test performance stays the same when the feature dimension is gradually reduced from 2048 to only 16 dimensions. This clearly validates the effectiveness of the proposed scheme.

4.3. Comparing with the state-of-the-art SemSL

In the following section, we proceed to demonstrate the performance of CRAE+, that is, the extended CRAE method by incorporating the two extensions discussed in Section 3.4. We compare its performance against the current state-of-the-art methods in SemSL. Similar to [2], several SemSL baselines are considered: Pseudo-Label, Π-Model, Mean Teacher, Virtual Adversarial Training (VAT) and MixMatch¹. Since a fair and comprehensive comparison has been done in [2], we strictly follow the same experimental setting and directly compare our proposed methods CRAE and CRAE+ to the numbers reported in [2].

The experimental results are shown in Figure 5, Figure 6 and Table 5. As seen from those Figures and Table, by incorporating the proposed two extensions, CRAE+ achieves significant improvement over the basic CRAE. Moreover, we find that the proposed CRAE+ is on-par with the best performed approaches, e.g., Mixmatch, in those datasets. This clearly demonstrates the power of the pro-

¹For CIFAR-100, we only compare CRAE+ against SWA [1] and MixMatch for their achieving the best reported performance in literature.

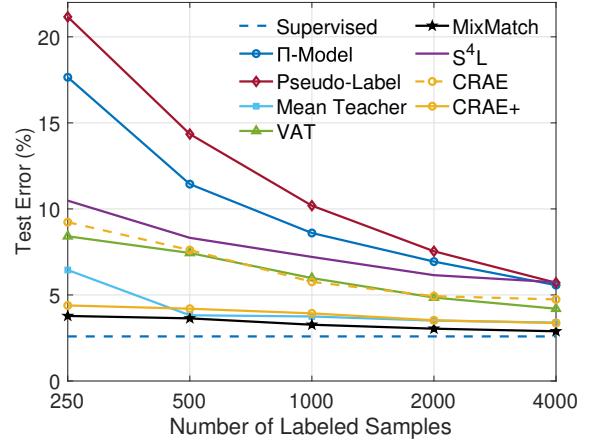


Figure 6. SVHN error rates (%) of CRAE+ and SOTA SemSL.

posed method. Note that the current state-of-the-art in SemSL is achieved by carefully combining multiple existing successful ideas in SemSL. In contrast, our CRAE+ achieves excellent results via an innovative framework of marrying SlfSL with SemSL. Conceptually, the latter enjoys greater potential. In fact, CRAE might be further extended by using more successful techniques in SemSL, such as mixup [19]. Since the focus of this paper is to study how SlfSL can benefit SemSL, we do not pursue this direction.

4.4. Ablation Study

Both CRAE and CRAE+ methods consist of multiple components. In this section, we conduct ablation studies to examine their impact and investigate how our methods work. Specifically, we first examine the importance of some key components in CRAE and CRAE+. Followed by an investigation on the relationship between CRAE and ensemble methods. Finally, we also study the effect of auxiliary semantic classifier to both CRAE and CRAE+ methods.

1. The impact of various components in CRAE and CRAE+.

We study these effects through adding or removing some components in order to provide additional insight into the role of each part in CRAE and CRAE+. Specifically, we measure the effect of (1) only adding extension 1 to CRAE, i.e., make the prediction of semantic classifier more sharpen. (2) further adding extension 2 to CRAE. The resulted model is identical to CRAE+. (3) removing whole rotation angle prediction branches from CRAE, i.e., pure supervised method with data rotated. (4) removing rotation angle prediction branches and adding extension 1 to CRAE. The resulted structure can be seen as a variant of only using the SemSL technique in Extension 1.

We conduct our studies on CIFAR-10 with 250 and 4000 labels with results presented in Table 6. The main observa-

Table 6. Ablation study to the impact of two extensions for CRAE on CIFAR-10 with 250 and 4000 labels.

| # Labels | 250 | 4000 |
|-------------------------------------|-------|-------|
| CRAE | 17.07 | 9.07 |
| CRAE + ext. 1 | 12.68 | 7.05 |
| CRAE + ext. 1 + ext. 2 (=CRAE+) | 10.76 | 5.98 |
| CRAE w/o rotation branches | 62.73 | 27.31 |
| CRAE w/o rotation branches + ext. 1 | 54.09 | 14.38 |

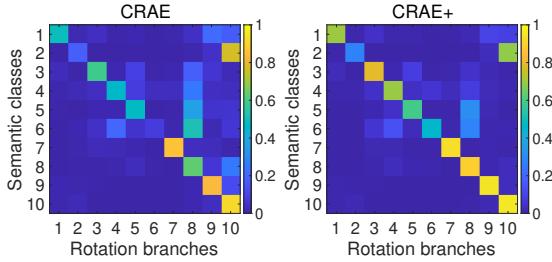


Figure 7. Comparison of rotation predictors’ diversity (based on the model at 1024 epoch) between CRAE and CRAE+ methods on CIFAR-10 with 250 labels.

tions are: (1) The two extensions in CRAE+ will bring varying degrees of improvement. In Figure 7, we also compare the confusion matrices (please see Figure 3 and section 3.3 for its definition) of CRAE and CRAE+. As seen, the confusion matrix of CRAE+ is closer to diagonal matrix, and it actually mitigates the confusion of some classes. This validates that CRAE+ indeed makes the conditional rotation estimator more specialized to its assigned class as expected. (2) Applying rotation as a data augmentation for labeled data will not lead to improved performance over the labeled-only baseline as by cross referring the results in Table 1. This shows that the advantage of CRAE is not coming from the rotation data augmentation. (3) Using the sharpening strategy as in our extension 1 and training a SemSL method alone (the last method in Table 6) does not produce good performance. This indicates the superior performance of CRAE+ is not simply coming from a strong SemSL method but its incorporation with the CRAE framework.

2. The relationship to ensemble method.

The structure of CRAE enjoys the benefit of ensembling rotation predictors. In order to investigate to what extent ensemble contributes to the good performance of CRAE, we compare CRAE with three ensemble baselines. The first one replaces the guidance from $p(y|x)$ with a multinomial random vector which means the rotation predictor is randomly selected. The second one independently trains all rotation estimation branches. The last one uses $p(y|x)$ to perform softly selection as in CRAE, but not back-propagates gradient to it. This operation is usually called “detach”.

From the results shown in Table 7, we can see that using

Table 7. Ablation study to the relationship between CRAE and ensemble methods on CIFAR-10 with 250 and 4000 labels.

| # Labels | 250 | 4000 |
|--------------------------|-------|-------|
| CRAE | 17.07 | 9.07 |
| Ensemble (random select) | 28.68 | 14.42 |
| Ensemble (independent) | 23.26 | 11.66 |
| CRAE detach $p(y x)$ | 23.27 | 11.88 |
| CRAE+ | 10.76 | 5.98 |

Table 8. Ablation study to the effect of additional semantic classifier to CRAE and CRAE+ methods.

| use of add. cls. | CRAE | | CRAE+ | |
|------------------|-------|-------|-------|-------|
| | ✓ | ✗ | ✓ | ✗ |
| CIFAR10-250 | 17.07 | 18.33 | 10.76 | 11.37 |
| CIFAR10-4000 | 9.07 | 10.25 | 5.98 | 6.28 |

ensemble strategy indeed leads to improved performance compared to S^4L method (numbers can be referred in Table 1). Among three ensemble methods, using independent branches and detach $p(y|x)$ achieves the best performance. However, CRAE still achieves superior performance over them, especially when the number of labeled samples is small. This clearly shows that CRAE can be benefited from the indirect supervision on $p(y|x)$ generated during optimizing $p(z|x)$ and the good performance of CRAE cannot be simply attributed to the ensemble strategy. Moreover, with the proposed extensions, CRAE+ achieves significantly better performance than the ensemble baselines.

3. The benefit of using additional semantic classifier.

As discussed in Section 3.5, adding an additional semantic classifier branch to CRAE can slightly improve the performance. In this subsection, we conduct an ablation study to examine its impact on both CRAE and CRAE+. The results are shown in Table 8. As seen, for CRAE, the additional classifier will bring more than one percent improvement on both 250 and 4000 labels of CIFAR-10 dataset. But for CRAE+, the additional classifier brings marginal improvement. This is perhaps due to that the two extensions make the supervision for $p(y|x)$ less noisy.

5. Conclusion

In this work, we propose an idea to couple SemSL with SfSL. Implementing this idea with rotation-angle-estimation-based SfSL, we design a new semi-supervised learning featured by conditional rotation angle estimation. Two extensions are further developed to enhance its performance. We show that the proposed method can achieve comparable performance to the state-of-the-art SemSL methods.

6. Appendix

6.1. Algorithm details of CRAE and CRAE+

The pseudocodes of CRAE and CRAE+ are presented in Algorithm 1 and Algorithm 2 respectively.

6.1.1 Experimental details

The experimental details are presented in Table 12.

6.1.2 Tabular results

Table 9 and 10 presents a summary for error rate comparison of CRAE and CRAE+ to existing SemSL methods on CIFAR-10 and SVHN respectively. Results in top block are reported in literature where mark \dagger means that the results come from [14], mark \ddagger means that the results come from [19] and others are from [2]. Results locating in the bottom block are achieved by our implementation.

Table 9. Error rate (%) comparison of CRAE to existing SemSL methods on CIFAR-10.

| # Labels | 250 | 500 | 1000 | 2000 | 4000 |
|------------------------|-------|-------|-------|-------|-------|
| Labeled-only \dagger | - | - | - | - | 20.26 |
| Π -Model | 53.02 | 41.82 | 31.53 | 23.07 | 17.41 |
| Pseudo-Label | 49.98 | 40.55 | 30.91 | 21.96 | 16.21 |
| Mixup | 47.43 | 36.17 | 25.72 | 18.14 | 13.15 |
| VAT | 36.03 | 26.11 | 18.68 | 14.40 | 11.05 |
| MeanTeacher | 47.32 | 42.01 | 17.32 | 12.17 | 10.36 |
| MixMatch | 11.08 | 9.65 | 7.75 | 7.03 | 6.24 |
| ICT \ddagger | - | - | 15.48 | 9.26 | 7.29 |
| Labeled-only | 56.76 | 47.24 | 36.09 | 29.90 | 19.65 |
| Fine-tune | 53.66 | 35.18 | 28.17 | 22.00 | 17.39 |
| S^4L | 32.66 | 28.23 | 22.55 | 18.91 | 15.71 |
| CRAE | 17.07 | 15.57 | 12.90 | 10.72 | 9.07 |
| CRAE+ | 10.76 | 9.20 | 7.43 | 6.98 | 5.98 |

6.2. Ablation study to the independent ensemble method

We further conduct an ablation study to the independent ensemble method to test its performance on various numbers of rotation classifiers. According to the results shown in Table 11, more rotation angle classifiers are limited to the promotion of the independent ensemble method.

References

- [1] Ben Athiwaratkun, Marc Finzi, Pavel Izmailov, and Andrew Gordon Wilson. There are many consistent explanations of unlabeled data: Why you should average. In *International Conference on Learning Representations*, 2019. 7

Table 10. Error rate (%) comparison of CRAE to existing SemSL methods on SVHN.

| # Labels | 250 | 500 | 1000 | 2000 | 4000 |
|------------------------|-------|-------|-------|-------|------|
| Labeled-only \dagger | - | - | 12.83 | - | - |
| Π -Model | 17.65 | 11.44 | 8.60 | 6.94 | 5.57 |
| Pseudo-Label | 21.16 | 14.35 | 10.19 | 7.54 | 5.71 |
| Mixup | 39.97 | 29.62 | 16.79 | 10.47 | 7.96 |
| VAT | 8.41 | 7.44 | 5.98 | 4.85 | 4.20 |
| MeanTeacher | 6.4 | 3.82 | 3.75 | 3.51 | 3.39 |
| MixMatch | 3.78 | 3.64 | 3.27 | 3.04 | 2.89 |
| ICT \ddagger | 4.78 | 4.23 | 3.89 | - | - |
| Labeled-only | 23.27 | 17.85 | 12.84 | 9.65 | 7.26 |
| Fine-tune | 16.21 | 11.54 | 8.31 | 7.30 | 5.61 |
| S^4L | 10.48 | 8.32 | 7.21 | 6.15 | 5.74 |
| CRAE | 9.23 | 7.61 | 5.76 | 4.93 | 4.74 |
| CRAE+ | 4.10 | 4.21 | 3.91 | 3.59 | 3.36 |

Table 11. Ablation study to the effect of rotation classifiers’ number on independent ensemble method.

| rot. cls. num. | 2 | 10 | 20 |
|----------------|-------|-------|-------|
| CIFAR10-250 | 23.29 | 23.26 | 23.04 |

- [2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, 2019. 1, 2, 5, 7, 9
- [3] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100. Citeseer, 1998. 2
- [4] Alexander Gammerman, Volodya Vovk, and Vladimir Vapnik. Learning by transduction. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pages 148–155. Morgan Kaufmann Publishers Inc., 1998. 2
- [5] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018. 1, 2, 3, 4
- [6] Thorsten Joachims. Transductive learning via spectral graph partitioning. In *Proceedings of the 20th International Conference on Machine Learning*, pages 290–297, 2003. 2
- [7] Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. Semi-supervised learning with deep generative models. In *Advances in neural information processing systems*, pages 3581–3589, 2014. 2
- [8] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *The IEEE Conference on Computer Vision and Pattern Recognition*, June 2019. 2, 6
- [9] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 5

Algorithm 1: CRAE Pseudocode

Inputs:

$\{X_l, Y_l\}$: collection of labeled samples
 X_u : collection of unlabeled samples
 f_{sem_cls} : semantic classifier
 $f_{rot_cls}^k$: k -th rotation angle classifier, where $k \in [1, 2, \dots, C]$
 T : total number of iterations
 B : minibatch size

Outputs:

f_{sem_cls} : semantic predictor for test set

Process:

- 1: **for** $t \leftarrow 1$ to T **do**
 - 2: Sample B examples from $\{X_l, Y_l\}$ and X_u respectively
 - 3: Obtain **semantic class prediction** on semantic classifier: $p(y|x_l) = f_{sem_cls}(x_l)$ and train with CrossEntropy($p(y|x_l), y_l$) loss
 - 4: Obtain **rotation angle prediction** on labeled samples $p(z|x_l) = \sum_{y_l} p(z|y_l, x_l) \cdot \text{one_hot}(y_l) = f_{rot_cls}^{y_l}(x_l)$ and train with CrossEntropy($p(z|x_l), \hat{z}$) loss // \hat{z} is rotation angle ground truth
 - 5: Obtain **rotation angle prediction** on unlabeled samples $p(z|x_u) = \sum_y p(z|y, x_u) \cdot p(y|x_u) = \sum_{k=1}^C f_{rot_cls}^k(x_u) \cdot f_{sem_cls}^k(x_u)$ and train with CrossEntropy($p(z|x_u), \hat{z}$) loss
 - 6: **end for**
-

Algorithm 2: CRAE+ Pseudocode

Inputs:

$\{X_l, Y_l\}$: collection of labeled samples
 X_u : collection of unlabeled samples
 f_{sem_cls} : semantic classifier
 $f_{rot_cls}^k$: k -th rotation angle classifier, where $k \in [1, 2, \dots, C]$
 T : total number of iterations
 B : minibatch size

Outputs:

f_{sem_cls} : semantic predictor for test set

Process:

- 1: **for** $t \leftarrow 1$ to T **do**
 - 2: Sample B examples from $\{X_l, Y_l\}$ and X_u respectively
 - 3: Obtain **semantic class prediction** on semantic classifier: $p(y|x_l) = f_{sem_cls}(x_l)$ and train with CrossEntropy($p(y|x_l), y_l$) loss
 - 4: Construct sharpened target \hat{p} for unlabeled data // see extension 1 in Section 3.4 of main text
 - 5: Obtain **semantic class prediction for unlabeled data** on semantic classifier $p(y_u|x_u) = f_{sem_cls}(x_u)$ and train with CrossEntropy($p(y_u|x_u), \hat{p}$) loss
 - 6: Construct mixed images $\{x_l^m, x_u^m\}$ from $\{X_l, X_u\}$ // see extension 2 in Section 3.4 of main text
 - 7: Obtain **rotation angle prediction** on labeled samples $p(z_m|x_l^m) = \sum_{y_l} p(z_m|y_l, x_l^m) \cdot \text{one_hot}(y_l) = f_{rot_cls}^{y_l}(x_l^m)$ and train with CrossEntropy($p(z_m|x_l^m), \hat{z}$) loss // \hat{z} is rotation angle ground truth
 - 8: Obtain **rotation angle prediction** on unlabeled samples $p(z_m|x_u^m) = \sum_y p(z_m|y, x_u^m) \cdot p(y|x_u) = \sum_{k=1}^C f_{rot_cls}^k(x_u^m) \cdot f_{sem_cls}^k(x_u)$ and train with CrossEntropy($p(z_m|x_u^m), \hat{z}$) loss
 - 9: **end for**
-

[10] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017. [1](#), [2](#)

[11] Ruihao Li, Sen Wang, Zhiqiang Long, and Dongbing Gu.

Undeepvo: Monocular visual odometry through unsupervised deep learning. In *The IEEE International Conference on Robotics and Automation*, pages 7286–7291, 2018. [1](#), [2](#)

[12] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and

Table 12. Experimental details.

| datasets | CIFAR-10 | SVHN/+Extra | CIFAR-100 | ILSVRC2012-1% | ILSVRC2012-10% |
|-------------------|------------------------------|---------------|---------------|---------------|----------------|
| architecture | WRN-28-2 | WRN-28-2 | WRN-28-8.4375 | ResNet50v2 | ResNet50v2 |
| # training set | 50000 | 73257/+531131 | 50000 | 1281167 | 1281167 |
| # labeled set | {250, 500, 1000, 2000, 4000} | | 10000 | 13762 | 128866 |
| # validation set | 5000 | 7325 | 5000 | 5005 | 50046 |
| minibatch size | 64 | 64 | 64 | 256 | 256 |
| optimizer | Adam | Adam | Adam | SGD | SGD |
| LR | 0.002 | 0.002 | 0.002 | 0.01 | 0.1 |
| weight decay | 0.02 | 0.02/0.0001 | 0.04 | 0.01 | 0.001 |
| # epoch | 1024 | 500 | 300 | 1000 | 200 |
| # iteration/epoch | 1024 | 1024 | 1024 | 53 | 503 |
| LR rampup | X | X | X | 10 epoch | 5 epoch |
| LR decay | X | X | X | 10 | 10 |
| LR decay at | X | X | X | {700,800,900} | {140,160,180} |
| EMA model | ✓ | ✓ | ✓ | X | X |

- Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *The IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979–1993, 2018. 2
- [13] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *Advances in Neural Information Processing Systems*, 2011. 5
- [14] Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. In *Advances in Neural Information Processing Systems*, pages 3235–3246, 2018. 9
- [15] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016. 2
- [16] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 5
- [17] Jost Tobias Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. In *International Conference on Learning Representations*, 2016. 2
- [18] 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*, pages 1195–1204, 2017. 1, 2
- [19] Vikas Verma, Alex Lamb, Juho Kannala, Yoshua Bengio, and David Lopez-Paz. Interpolation consistency training for semi-supervised learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 3635–3641, 7 2019. 2, 7, 9
- [20] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S⁴l: Self-supervised semi-supervised learning. In *The IEEE International Conference on Computer Vision*, October 2019. 1, 2, 5, 6
- [21] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 1, 2, 5
- [22] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European Conference on Computer Vision*, pages 649–666, 2016. 2
- [23] Zhi-Hua Zhou and Ming Li. Tri-training: Exploiting unlabeled data using three classifiers. *IEEE Transactions on Knowledge & Data Engineering*, (11):1529–1541, 2005. 2