Mutual Learning of Complementary Networks via Residual Correction for Improving Semi-Supervised Classification

Si Wu[12]   Jichang Li[1]   Cheng Liu[2]   Zhiwen Yu[1]   Hau-San Wong[2]

[1]School of Computer Science and Engineering, South China University of Technology

[2]Department of Computer Science, City University of Hong Kong

cswusi@scut.edu.cn, cslijichang@mail.scut.edu.cn, cliu272-c@my.cityu.edu.hk

zhwyu@scut.edu.cn, cshswong@cityu.edu.hk

Abstract

Deep mutual learning jointly trains multiple essential networks having similar properties to improve semi-supervised classification. However, the commonly used consistency regularization between the outputs of the networks may not fully leverage the difference between them. In this paper, we explore how to capture the complementary information to enhance mutual learning. For this purpose, we propose a complementary correction network (C-CN), built on top of the essential networks, to learn the mapping from the output of one essential network to the ground truth label, conditioned on the features learnt by another. To make the second essential network increasingly complementary to the first one, this network is supervised by the corrected predictions. As a result, minimizing the prediction divergence between the two complementary networks can lead to significant performance gains in semi-supervised learning. Our experimental results demonstrate that the proposed approach clearly improves mutual learning between essential networks, and achieves state-of-the-art results on multiple semi-supervised classification benchmarks. In particular, the test error rates are reduced from previous 21.23% and 14.65% to 12.05% and 10.37% on CIFAR-10 with 1000 and 2000 labels, respectively.

1. Introduction

One of the main limitations of applying deep convolutional networks [16] [36] [12] is the need for massive collection of labeled images. To bypass expensive manual annotations, many studies have been performed on semi-supervised learning [21] [5] [40] [2], such that the models can be trained on partially labeled data, since it is more practical to expect that only a small fraction of samples can receive human annotations. In order to use unlabeled data to improve the generalization capability of classifiers,
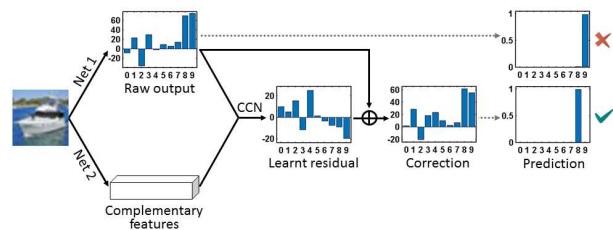

Figure 1. An example to illustrate how the proposed CCN improves semi-supervised classification on CIFAR-10 with 1000 labels (classes 0-9 denote 'plane', 'auto', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship' and 'truck', respectively). CCN learns the mapping from the raw output (input of the softmax layer) of one network (Net 1) to the ground truth label, conditioned on the features learnt by another network (Net 2). According to the raw output, the 'ship' image is misidentified as 'truck'. CCN is able to produce a compensatory residual to correct the misclassification.

semi-supervised methods rely on an important assumption that it is more likely for neighboring data points to belong to the same class, which means that the decision boundaries should be located in low-density regions. There are many deep models that have been developed based on this assumption, such as [35] [25] [34]. The prediction of a classifier should be consistent on the unlabeled data irrespective of whether perturbations have been added. Previous methods including Temporal Ensembling [18], virtual adversarial training (VAT) [26] and adversarial dropout (VAdD) [29] follow similar principles. On the other hand, mutual learning between separate networks is also effective for determining more reliable decision boundaries. Recent methods, such as dual learning [11], Mean-Teacher [38] and deep mutual learning (DML) [42], have brought improvement in semi-supervised classification. Most of them penalize inconsistent predictions of different networks on unlabeled data. However, these methods only consider the difference between them, while ignoring the complementarity.

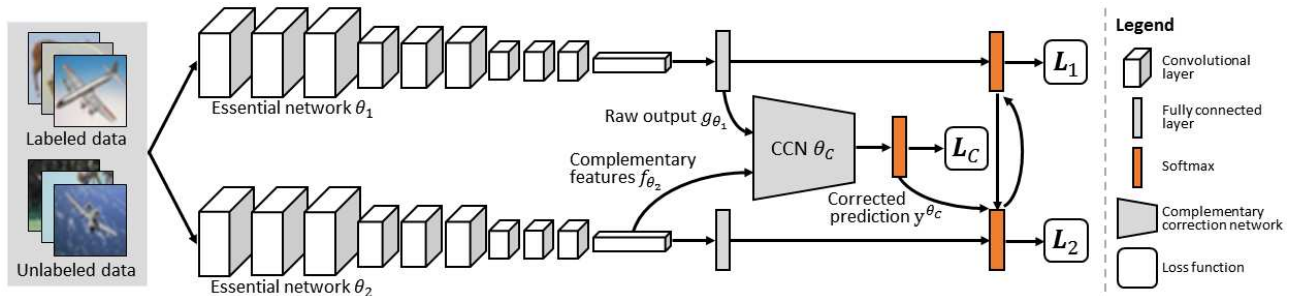We are concerned with the task of improving mutual

Figure 2. An overview of our enhanced mutual learning model for semi-supervised classification. Our model consists of two essential networks having similar properties and a complementary correction network (CCN). CCN learns to more accurately classify the unlabeled instances, conditioned on the output of one network and the features of the other network. The second essential network is supervised by the output of CCN, and becomes increasingly complementary to the first one as it learns. The resulting essential networks lead to significant performance gains due to complementary knowledge transfer via mutual learning.

learning for semi-supervised classification. To fully utilize the complementary information contained in the different networks, we aim to learn the mapping from the output of one network to the ground truth label, conditioned on the feature learnt by another, as shown in Figure 1. This mapping not only learns the prediction deviation, but also helps in improving classification on unlabeled data. The resulting better predictions provide further guidance to the training of complementary networks, such that mutual learning between these networks is able to bring significant performance gains.

In this paper, we present an enhanced mutual learning approach to train complementary networks for improving semi-supervised classification. Specifically, we extend the DML model by including a complementary correction network (CCN) to capture complementary information between two essential networks. This new network is built on top of the essential networks, and is conditionally dependent on the raw output (input of the softmax layer) of one network and the features provided by another. We adopt a residual architecture, such that CCN is able to learn the difference between the raw output and ground truth label conditioned on the input features. As a result, more accurate classification on unlabeled data is produced and utilized to train the second essential network, which in turn becomes more discriminative and complementary to the first one as it learns. By minimizing the divergence between these two essential networks, the knowledge learnt by CCN can be ultimately transferred to the first one, and lead to additional performance gains. An overview of the proposed approach is shown in Figure 2. In the experiments, we present state-of-the-art results achieved on multiple standard semi-supervised classification benchmarks, and insights on why the proposed approach works.

This work makes the following contributions. (1) Instead of directly minimizing the prediction divergence between separate networks, we propose the CCN to significantly improve semi-supervised mutual learning, by capturing and transferring complementary knowledge between the networks. (2) CCN is able to use the learnt features of one network to help correct the outputs of the other network. The resulting more accurate classification on unlabeled data is further leveraged to guide model training, such that the networks become increasingly complementary as they learn. (3) We demonstrate that the proposed enhanced mutual learning model is more effective than the DML model, and improves the state-of-the-art results on multiple standard semi-supervised learning benchmarks.

2. Related Work

We restrict our review to the closely related work, especially the recent advances in semi-supervised learning using deep models. To aid classifiers to explore the categories of unlabeled data, Generative Adversarial Networks (GANs) [8] [30] [24] have been applied to semi-supervised learning, such as [14] [28] [17]. In [37], Springenberg proposed a categorical GAN to regularize a discriminatively trained classifier, such that a robust classification model can be achieved. In [33], Salimans et al. explored various practical techniques for improving the training of generative models and semi-supervised classification. Furthermore, Wei et al. [39] improved the training of Wasserstein GANs [1] by including a consistency regularization to the discriminator, such that the Lipschitz continuity can be enhanced and promising results are achieved. To characterize the class-conditional distributions, Li et al. [20] proposed a triple generative adversarial network to include a classifier in a three-player formulation. Another similar work reported in [7] presented a triangle GAN framework, in which two generators and two discriminators are employed to characterize the joint distribution of instances and labels. In contrast to the above GAN-based methods which aim to generate images as good as possible, the GAN in [4] generates 'bad' images which are located at the low density regions and

thus may be close to the decision boundaries in the latent space, based on which the discriminative capability of the classifier can be improved.

The perturbation-based models have shown promising results through introducing noise to model training for reducing overfitting, such as [31] [32]. In [18], the training is performed by penalizing the difference between the predictions of the network with and without stochastic augmentation, such that the smoothness in the output of the network with respect to the input is encouraged. Similar to adversarial training [9], Miyato et al. [26] [25] proposed a virtual adversarial training method to select the perturbations in the direction sensitive to the prediction of the classifier. From another perspective, adversarial dropout [29] was proposed to generate the perturbation to model updating by maximizing the divergence between the predicted class distribution and ground truth label.

Mutual learning is another effective strategy for improving semi-supervised learning. To acquire training experience from another network, distillation based methods [13] were proposed to train a separate and relatively small network. Different from distillation, mutual learning starts with a set of essential networks, which jointly learn to solve the tasks. In [3], Batra and Parikh proposed a cooperative learning paradigm to jointly train multiple models specializing to different domains, and learn domain-invariant visual attributes. In [42], Zhang et al. proposed a deep mutual learning model which minimizes the divergence between the outputs of two networks having different parameter initializations and dropout. To construct a better teacher model for enhancing mutual learning, Tarvainen and Valpola [38] adopted the exponential moving average of a student network as a teacher to provide training targets for the student.

There are substantial differences between our proposed framework and existing works. The main difference is in the way the models are learnt. We propose the CCN which is built on top of two essential networks. Its main role is to learn to correct the output of one network, and guide the training of the other network. As a result, the complementarity between the essential networks can be significantly enhanced. To our best knowledge, there have been no previous attempts to capture the complementary information between separate networks for enhancing mutual learning, in the way that our CCN is designed to do.

3. Proposed Approach

The semi-supervised setting naturally occurs for cases in which a large number of images can be easily collected from the web but only a small portion of them are manually labeled. In our problem, we consider that the training set $\mathcal{X} = \mathcal{L} \cup \mathcal{U}$ contains N instances, out of which the subset $\mathcal{L} = \{(x_i, y_i)\}_{i=1}^{N_{\mathcal{L}}}$ is labeled and the remainder $\mathcal{U} = \{x_j\}_{j=1}^{N_{\mathcal{U}}}$ is unlabeled, where (x_i, y_i) denotes a labeled instance and the corresponding class label, and x_j denotes an unlabeled instance. In the semi-supervised setting, we have $N_{\mathcal{L}} \ll N_{\mathcal{U}}$.

Deep mutual learning models usually consist of two or more essential networks. Since deep convolutional networks for image classification have high capacity, jointly training two networks can achieve a trade-off between performance gains and computational cost in most cases. Here we introduce a dual-net based mutual learning model. Specifically, we design a CCN, parameterized by θ_C, to leverage the complementary information between the two essential networks parameterized by θ_1 and θ_2, respectively. CCN can be expected to produce more accurate classification on unlabeled data, and guide the training of complementary essential networks in our model.

3.1. Enhanced Mutual Learning Model

We extend the DML model by including a CCN to leverage complementary information from essential networks. CCN has two separate inputs, the raw output of one essential network, as well as the learnt features of the other essential network for modeling the divergence between the raw output and ground truth label. Compared to the first network, CCN is able to produce more accurate classification on unlabeled data, which can be utilized for guiding the training of the second network. By minimizing the divergence between the two essential networks, both of them can be further improved.

Specifically, the overall loss function $\boldsymbol{L}_1$ for the first essential network is composed of the following four terms:

$$
\begin{aligned}
\boldsymbol{L}_1(\theta_1; \mathcal{X}) = &\sum_{(x_i, y_i) \in \mathcal{L}} \ell\big(y_i, h_{\theta_1}(x_i)\big) + \sum_{x_j \in \mathcal{U}} \mathcal{H}\big(h_{\theta_1}(x_j)\big) \\
&+ \eta \sum_{x_j \in \mathcal{U}} \mathcal{A}(\theta_1; x_j) + \lambda \sum_{x_j \in \mathcal{U}} \mathcal{D}_{KL}\big(h_{\theta_2}(x_j) \| h_{\theta_1}(x_j)\big),
\end{aligned} \tag{1}
$$

where $h_{\theta_1}(\cdot)$ $(h_{\theta_2}(\cdot))$ denotes the predicted class probability distribution of the network θ_1 (θ_2) for an input, $\ell(\cdot, \cdot)$ denotes the cross-entropy function, $\mathcal{H}(\cdot)$ denotes the conditional entropy function with respect to the posterior class probability distribution, $\mathcal{A}$ denotes a perturbation-based virtual adversarial training term, and $\mathcal{D}_{KL}(\cdot \| \cdot)$ denotes the Kullback-Leibler (KL) divergence between two different distributions. The coefficients η and λ are the weighting factors for achieving a balance among the terms in $\boldsymbol{L}_1$. In Eq.(1), $\mathcal{H}(\cdot)$ is used to quantify the amount of information needed to describe the class label of an unlabeled instance according to the network prediction as follows:

$$
\mathcal{H}\big(h_{\theta_1}(x_j)\big) = -h_{\theta_1}(x_j)^T \ln h_{\theta_1}(x_j). \tag{2}
$$

Minimization of the conditional entropy term enhances the confidence of the classifier on unlabeled instances, which in turn drives the decision boundaries away from data-dense

regions to facilitate semi-supervised learning, as pointed out in [10] [26]. To stabilize the estimation of the conditional entropy on the unlabeled instances, $\mathcal{A}$ is used to smooth the classifier with respect to input perturbations as follows:

$$\mathcal{A}(\theta_1; x_j) = \max_{\|\nu\| \le \epsilon} \mathcal{D}_{KL}\big(h_{\theta_1}(x_j) \| h_{\theta_1}(x_j + \nu)\big), \quad (3)$$

where ϵ denotes a hyper-parameter controlling the intensity of the adversarial perturbation ν. Furthermore, minimizing the last term in $\boldsymbol{L}_1$ encourages the two networks to produce consistent predictions. In fact, this mutual learning term is important for providing training experience in the form of predicted class distributions on unlabeled instances.

In addition, the overall loss function $\boldsymbol{L}_2$ for the second essential network is defined as follows:

$$\begin{aligned}
\boldsymbol{L}_2(\theta_2; \mathcal{X}) = &\sum_{(x_i, y_i) \in \mathcal{L}} \ell\big(y_i, h_{\theta_2}(x_i)\big) + \sum_{x_j \in \mathcal{U}} \mathcal{H}\big(h_{\theta_2}(x_j)\big) \\
&+ \sum_{x_j \in \mathcal{U}} \ell\big(\mathrm{y}_j^{\theta_C}, h_{\theta_2}(x_j)\big) + \eta \sum_{x_j \in \mathcal{U}} \mathcal{A}(\theta_2; x_j) \\
&+ \lambda \sum_{x_j \in \mathcal{U}} \mathcal{D}_{KL}\big(h_{\theta_1}(x_j) \| h_{\theta_2}(x_j)\big),
\end{aligned} \quad (4)$$

where $\mathrm{y}_j^{\theta_C}$ denotes the pseudo label of instance x_j according to the prediction of CCN (to be introduced in detail in the next subsection). Note that the two essential networks are trained under different supervision. Different from the first one, the second network learns to predict class labels of unlabeled instances by imitating the outputs of CCN as ground truth targets. As a result, the second network becomes increasingly complementary to the first one, since it should have similar classification performance with CCN.

3.2. Complementary Correction Networks

We propose CCN to leverage the complementary information from essential networks to produce more accurate predictions. This network learns a mapping from the output of one essential network to the ground truth label, conditioned on the high level features of the other essential network. Inspired by the work of He et al. [12], an important feature of our CCN is an identity-skip connection, which adds the raw output of the first essential network to the end of this correction module. This skip connection is different from the residual network, due to the reason that we take into account the learnt features of the second essential network as side input, and thus our correction network is able to capture the complementary information.

As shown in Figure 3, the raw output of the first network is projected into a higher dimensional embedding. The abstract features of the second network are similarly projected into a lower dimensional embedding. To combine these two modalities, we concatenate the two embedding vectors, and feed the resulting vector to two fully connected layers, such that the vector is projected back into a valid label space. To
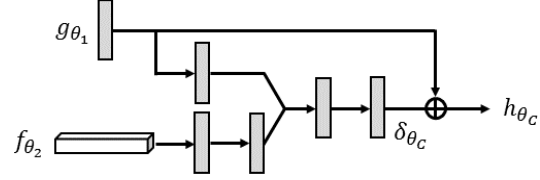

Figure 3. Illustration of the proposed CCN.

Table 1. The architecture of the CCN used in the proposed enhanced mutual learning model.

Layer	Description	
Input	Raw output of Net 1	Features of Net 2
$L-4$	Fully connected 10→64, LReLU	Fully connected 128→64, LReLU
$L-3$		Fully connected 64→64, LReLU
$L-2$	Concatenation, Fully connected 128→32, LReLU	
$L-1$	Fully connected 32→10	
$L-0$	Addition to the raw output, Softmax	

formulate the overall loss function of CCN, we adopt the cross entropy function as a classification term to capture the difference between the predicted and ground truth labels. In addition, we choose the mean square distance to measure the difference between the current and temporal ensemble predictions as follows:

$$\begin{aligned}
\boldsymbol{L}_C(\theta_1, \theta_2, \theta_C; \mathcal{X}) = &\sum_{(x_i, y_i) \in \mathcal{L}} \ell\big(y_i, h_{\theta_C}(x_i)\big) \\
&+ \mu \sum_{x_j \in \mathcal{U}} \big\| h_{\theta_C}(x_j) - \tau_j^{\theta_C} \big\|^2,
\end{aligned} \quad (5)$$

where $\tau_j^{\theta_C}$ denotes the temporal ensemble prediction of CCN to the label of instance x_j over previous training epochs. Since there are only a small number of labeled samples, the majority of training samples are unlabeled and may dominate the overall loss of CCN. Similar to [18], we use a ramp-up coefficient μ for the second term at the beginning to avoid this dominance. In our model, the temporal ensemble predictions are the following exponential moving averages of label predictions

$$\tau_j^{\theta_C} \leftarrow \alpha \tau_j^{\theta_C} + (1 - \alpha) h_{\theta_C}(x_j). \quad (6)$$

In each training epoch, the output of the network is accumulated into a temporal ensemble output, and a momentum coefficient α is used to control the extent of ensembling in the temporal dimension. Aggregating the previous predictions is expected to be more accurate.

Since CCN learns the mapping from the raw output of the first essential network to the ground truth label, the corrected class probability distribution can be computed as follows:

$$h_{\theta_C}(x_j) = \mathcal{N}\Big(g_{\theta_1}(x_j) + \delta_{\theta_C}\big(g_{\theta_1}(x_j), f_{\theta_2}(x_j)\big)\Big), \quad (7)$$

where $\mathcal{N}(\cdot)$ denotes the normalized exponential function, $g_{\theta_1}(\cdot)$ denotes the raw output of the first essential network, $f_{\theta_2}(\cdot)$ denotes the learnt representation on the global pooling layer of the second essential network, and $\delta_{\theta_C}(\cdot, \cdot)$ denotes the residual learnt by CCN.

To make the second essential network complementary to the first one, the prediction of CCN can be used to produce the training target of the second essential network. Specifically, $h_{\theta_C}(x_j) = [\text{h}_{j;1}, \text{h}_{j;2}, \ldots, \text{h}_{j;M}]$ is transformed to an one-hot vector $\text{y}_j^{\theta_C} = [\text{y}_{j;1}, \text{y}_{j;2}, \ldots, \text{y}_{j;M}]$ as a pseudo label of instance x_j as follows:

$$\text{y}_{j;m} = \begin{cases} 1, & \text{if } \text{h}_{j;m} = \arg\max_l \left[h_{\theta_C}(x_j)\right]_l, \\ 0, & \text{otherwise}, \end{cases} \quad (8)$$

where M denotes the number of classes, and $[\cdot]_l$ denotes the l-th component of the predicted class probability vector, indicating the probability of an instance belonging to the l-th class.

The architecture of our CCN is shown in Table 1. In the training process, an input image is processed by the essential networks to compute high-level image features and produce class probability predictions. Then, the raw output of the first network and the features learnt by the second network pass through CCN. The prediction of CCN is transformed into the pseudo label of the unlabeled instance with respect to the second network, but cannot incur gradients propagated back to itself. The implementation details of our proposed model are summarized in Algorithm 1.

4. Experiments and Discussion

In this section, we perform extensive experiments to verify the effectiveness of the proposed enhanced mutual learning model for improving semi-supervised classification. Specifically, we first evaluate our proposed approach, and then compare with the state-of-the-art methods on multiple semi-supervised learning benchmarks. For better understandings of our work, we also investigate the effectiveness of our proposed CCN and enhanced mutual learning mechanism through ablation studies and visualization.

4.1. Experimental Settings

We highlight the effectiveness of our CCN with a toy example, and then evaluate the proposed approach on the MNIST [19], SVHN [27], CIFAR-10 and CIFAR-100 [15] benchmarks, on which existing state-of-the-art methods for semi-supervised classification mostly focus. We report the average classification error and the corresponding standard deviation over 10 runs on the test data.

Model Variants. We build the following variants of our proposed model to assess the effectiveness of the improvement strategies to the final classification performance.

Algorithm 1 Pseudo-code of our enhanced mutual learning model for training two essential networks and CCN.

1: **Input:** Labeled data $(x_i, y_i) \in \mathcal{L}$ and unlabeled data $x_j \in \mathcal{U}$, weights η, λ and μ, and number of training epochs T.
2: **Initialize:** Essential networks θ_1 and θ_2, CCN θ_C, temporal ensemble predictions $\tau_j^{\theta_C}$ and pseudo labels $y_j^{\theta_C}$ of unlabeled samples, and learning rate γ.
3: **for** $t = 1$ to T **do**
4:     Randomly sample mini-batches from $\mathcal{L}$ and $\mathcal{U}$.
5:     **for** each mini-batch B **do**
6:         Compute the raw outputs $g_{\theta_1}(x_i)$ and $g_{\theta_1}(x_j)$, and evaluate the first essential network $h_{\theta_1}(x_i)$ and $h_{\theta_1}(x_j)$.
7:         Compute the features $f_{\theta_2}(x_i)$ and $f_{\theta_2}(x_j)$, and evaluate the second essential network $h_{\theta_2}(x_i)$ and $h_{\theta_2}(x_j)$.
8:         Evaluate CCN $h_{\theta_C}(x_i)$ and $h_{\theta_C}(x_j)$.
9:         Compute $y_j^{\theta_C}$ according to Eq.(8).
10:         Apply stochastic gradient descent and update $\theta_C \leftarrow Adam\big(\nabla_{\theta_C} \boldsymbol{L}_C(\theta_1, \theta_2, \theta_C; B), \theta_C, \gamma\big)$.
11:         Apply stochastic gradient descent and update $\theta_1 \leftarrow Adam\big(\nabla_{\theta_1}\big(\boldsymbol{L}_C(\theta_1, \theta_2, \theta_C; B) + \boldsymbol{L}_1(\theta_1; B)\big), \theta_1, \gamma\big)$.
12:         Apply stochastic gradient descent and update $\theta_2 \leftarrow Adam\big(\nabla_{\theta_2}\big(\boldsymbol{L}_C(\theta_1, \theta_2, \theta_C; B) + \boldsymbol{L}_2(\theta_2; B)\big), \theta_2, \gamma\big)$.
13:         Update $\tau_j^{\theta_C}$ according to Eq.(6).
14:     **end for**
15: **end for**
16: **Return** θ_1, θ_2 and θ_C.

'Baseline'. We train two essential networks having the same architecture as the proposed model by adopting the DML model [42]. The 'Baseline' results serve as the lower bound for our evaluation.

'Our Model w/o ML'. We disable mutual learning between essential networks, by removing the divergence term of their predictions from the corresponding loss functions, to analyze the capability of CCN in correcting the prediction of the first essential network.

'Our Model w/o CCN'. We remove the CCN from our model to investigate its effectiveness in exploiting the complementary information for enhancing mutual learning between the essential networks.

'Our Model w/o VAT'. We remove the divergence term of virtual adversarial training from the loss functions of the essential networks to train another variant of our model, such that we can investigate the complementary of our model with the existing technique [26].

4.2. Toy Example

To highlight the effectiveness of our CCN, we test the variant 'Our Model w/o ML' on the well-known 'two-spirals' synthetic dataset. We generate 1000 data points per class, and there are a total of 40 labeled data points. We adopt two essential networks consisting of 3 hidden layers of size 300 nodes with ReLU, and a corresponding CCN in our model. In Figure 4, we visualize the learnt decision boundaries during training to illustrate how the CCN corrects the predictions of the first essential network.

Table 2. Test error rates (%) of our models and the previous state-of-the-art methods on the MNIST, SVHN and CIFAR-10 datasets. The proposed approach achieves more accurate classification than the competing methods in all the cases.

| Method	MNIST		SVHN		CIFAR-10		
	50 labels	100 labels	500 labels	1000 labels	1000 labels	2000 labels	4000 labels
LadderNetwork[31]	-	1.06±0.37	-	-	-	-	20.40±0.47
CatGAN[37]	-	1.39±0.28	-	-	-	-	19.58±0.58
Improved GAN[33]	2.21±1.36	0.93±0.07	-	8.11±1.30	-	19.61±2.09	18.63±2.32
ALI[6]	-	-	-	7.42±0.65	-	-	17.99±1.62
TripleGAN[20]	1.56±0.72	0.91±0.58	-	5.77±0.17	-	-	16.99±0.36
GoodBadGAN[4]	-	0.80±0.10	-	4.25±0.03	-	-	14.41±0.03
SPCTN[41]	1.72±0.13	1.00±0.11	9.79±1.24	7.37±0.30	-	17.99±0.50	14.17±0.27
Π-model[18]	1.02±0.37	0.89±0.15	6.65±0.53	4.82±0.17	31.65±1.20	17.57±0.44	12.36±0.31
Temporal-Ensembling[18]	-	-	5.12±0.13	4.42±0.16	23.31±1.01	15.64±0.39	12.16±0.24
Mean-Teacher[38]	-	-	4.18±0.27	3.95±0.19	-	15.73±0.31	12.31±0.28
VAT[26]	-	-	-	3.74±0.09	-	-	11.96±0.10
VAdD[29]	-	-	-	4.16±0.08	-	-	11.68±0.19
VAdD+VAT[29]	-	-	-	3.55±0.05	-	-	10.07±0.11
SNTG+Π-model[22]	0.94±0.42	0.66±0.07	4.52±0.30	3.82±0.25	21.23±1.27	14.65±0.31	11.00±0.13
SNTG+VAT[22]	-	-	-	3.83±0.22	-	-	9.89±0.34
CT-GAN[39]	-	0.89±0.13	-	-	-	-	9.98±0.21
Baseline	8.48±1.03	3.47±0.67	15.03±0.11	10.74±0.10	29.57±0.89	20.97±0.37	15.33±0.31
Our Model	**0.67±0.13**	**0.42±0.11**	**3.63±0.21**	**3.36±0.18**	**12.05±0.42**	**10.37±0.31**	**8.80±0.24**
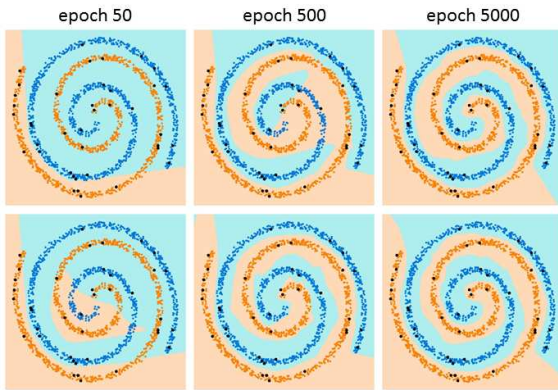
Figure 4. Comparison between the first essential network (upper row) and CCN (bottom row) in 'Our Model w/o ML' during training on the synthetic dataset. The labeled data points are marked black. Different colors indicate different classes. CCN efficiently converges to a better solution than the first network.

4.3. Comparison on Benchmarks

Comparison to Previous Work. We first report the results of the proposed approach, and perform a comparison with the existing state-of-the-art semi-supervised learning methods on the MNIST, SVHN and CIFAR-10 benchmarks. Table 2 shows the results of our model and the competing methods on these benchmarks for the cases where different number of labels are given. For a fair comparison, we evaluate the first essential network of our model in the test phase, instead of the ensemble of the essential networks, although there are three well-trained networks available at the end of training process. Compared to the competing methods, 'Our Model' achieves the best results in all the cases. In par-

Table 3. Test error rates (%) of our model and the variants on the CIFAR-10 dataset.

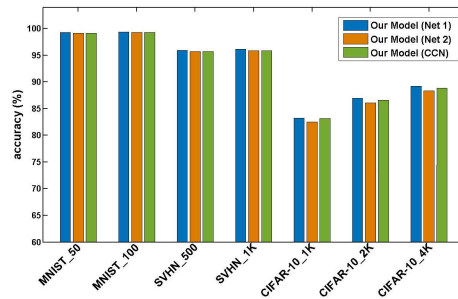| Method	1000 labels	2000 labels	4000 labels
Baseline	29.57±0.89	20.97±0.37	15.33±0.31
Our Model w/o ML	19.71±0.86	14.59±0.75	11.50±0.42
Our Model w/o CCN	20.41±0.42	13.34±0.27	11.45±0.22
Our Model w/o VAT	16.74±0.19	13.06±0.20	10.54±0.18
Our Model	**12.05±0.42**	**10.37±0.31**	**8.80±0.24**

Figure 5. Comparison of the two essential networks and CCN in our model on the MNIST, SVHN and CIFAR-10 datasets. The three networks achieve very similar performance in all cases due to complementary knowledge transfer during mutual learning.

ticular, the test error rate of the proposed approach reaches 12.05% and 10.37% on CIFAR-10 with 1000 and 2000 labels, which are lower than those of the second best method 'SNTG+Π-model' (21.23% and 14.65%) by about 9.2 and 4.3 percentage points, respectively. It is noted that 'Our Model' outperforms the previous state-of-the-art methods by a large margin.

Comparison to Model Variants. To confirm the effectiveness of the proposed approach, we also report the re-
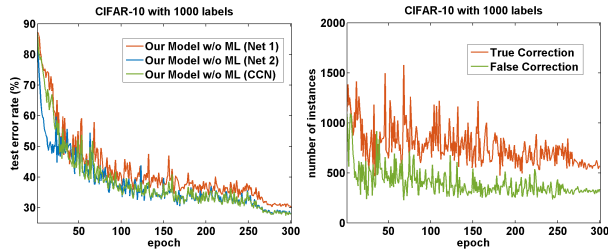
Figure 6. An example to illustrate the effectiveness of CCN on CIFAR-10 with 1000 labels. In the left subfigure, CCN outperforms the first essential network. In the right subfigure, the amount of true corrections is much greater than that of false corrections, which indicates that CCN is able to produce more accurate classification on unlabeled data.

sults of 'Baseline' on the benchmarks. Table 2 shows that 'Our Model' significantly outperforms 'Baseline' in all the cases. On MNIST with 50 labels, SVHN with 500 labels and CIFAR-10 with 1000 labels, the test error rates are reduced from 8.48%, 15.03% and 29.57% to 0.67%, 3.63% and 12.05%, and the corresponding performance gains are 7.8, 11.4 and 17.5 percentage points, respectively. Since the essential networks in 'Baseline' have the same architecture as the networks in 'Our Model', we consider that our CCN and enhanced mutual learning mechanism lead to the significant performance gains. To investigate the relative contributions of the improvement strategies, we perform a comparison between our model and the variants on CIFAR-10, and Table 3 shows that removing the corresponding terms leads to a significant drop in performance. We consider that CCN is important in facilitating mutual learning, and additional performance gains can be achieved by incorporating perturbation-based adversarial training.

4.4. Model Analysis

To provide insights on why the proposed approach works, we investigate how the proposed CCN and enhanced mutual learning mechanism improve the classification performance of the final model in the following four aspects.

Comparison of Member Networks. Our enhanced mutual learning model consists of three networks: two essential networks ('Net 1' and 'Net 2') and CCN. We compare these networks on all the three benchmarks. Figure 5 shows the average test error rates of the three networks for the different cases. One can observe that the three networks have very similar performance. This phenomenon is consistent with the characteristics of mutual learning. The second essential network learns to mimic CCN, and transfers the learnt knowledge to the first essential network by minimizing their prediction divergence.

Effectiveness of CCN. To verify the capability of our CCN in correcting the raw output of the first essential network, we compare the three networks of the variant 'Our
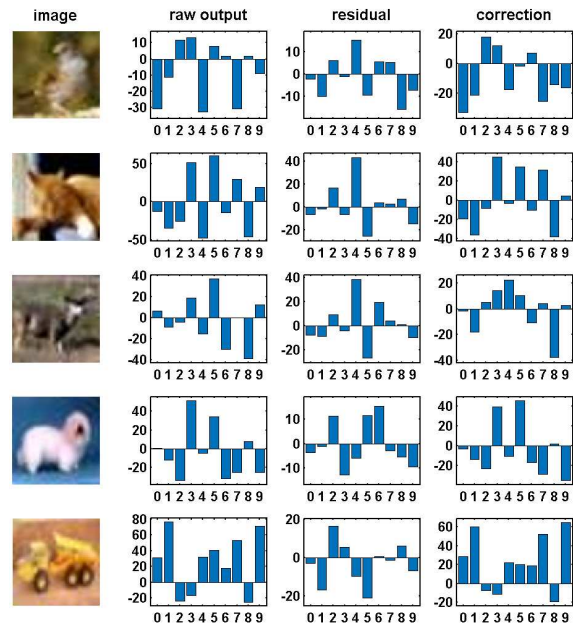

Figure 7. Representative results of CCN correcting the raw output of the first essential network on CIFAR-10 with 1000 labels (classes 0-9 denote 'plane', 'auto', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship' and 'truck', respectively). Although these images are misclassified according to the raw outputs, compensatory residuals can be learnt by CCN such that these misclassifications can be corrected.

Model w/o ML' in Figure 6. The left subfigure shows the performance of these three networks on CIFAR-10 with 1000 labels. Since CCN learns the residual between the raw output and ground truth label by exploiting the complementary information from the second essential network, it performs better than the first essential network. In addition, the second essential network is supervised by the output of CCN, and thus these two networks have very similar performance. In the right subfigure, we plot the numbers of the test instances on which the outputs of the first essential network are truly corrected and falsely corrected, respectively. The result shows that the amount of true corrections is much greater than that of false corrections, which indicates that our CCN does improve classification on unlabeled data by utilizing the complementary information between essential networks. Some representative corrections are visualized in Figure 7.

Effectiveness of Enhanced Mutual Learning. In our model, CCN contributes to forming a teacher by guiding the training of the second essential network, and transferring the knowledge to the first essential network via mutual learning with the second essential network. To demonstrate the superiority of the proposed model, Figure 8 shows the performance improvement of 'Our Model' over 'Baseline' during the training on SVHN with 500 labels and CIFAR-
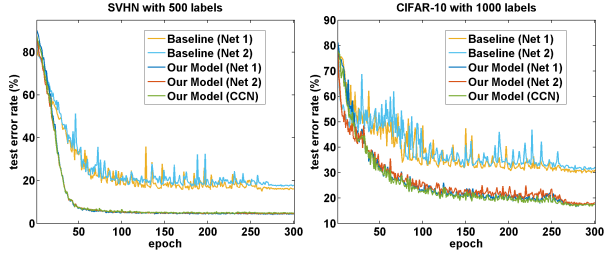
Figure 8. Comparison of the baseline model and our model on SVHN with 500 labels (left) and CIFAR10 with 1000 labels (right). Compared to the two networks of 'Baseline', the three networks of 'Our Model' consistently and efficiently converge to better solutions during training, which verifies the effectiveness and superiority of the proposed approach in mutual learning.
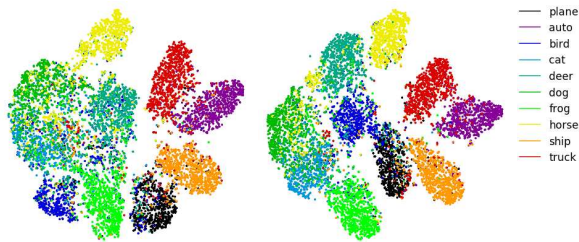

Figure 9. The t-SNE plot of the last hidden layer on the test data of CIFAR-10 with 1000 labels: the baseline model (left) and our model (right). Our model can learn more discriminative representations on which separating the data points of the difficult classes including 'cat', 'deer' and 'dog' becomes easier.

Table 4. Test error rates (%) of our model and the previous state-of-the-art methods on the CIFAR-100 dataset.

Method	5000 labels	10000 labels
Π-model[18]	-	39.19±0.36
Temporal Ensembling[18]	-	38.65±0.51
SNTG+Π-model[22]	-	37.97±0.29
Baseline	53.58±0.45	40.83±0.29
Our Model	**43.42±0.31**	**35.28±0.23**

10 with 1000 labels. In contrast to 'Baseline' which only penalizes the prediction divergence between essential networks, CCN explores more information from both essential networks, and is thus able to improve the prediction quality. Furthermore, the second essential network is able to learn better abstract representations by using the corrected prediction. In turn, the second network contributes to the performance gains of the first network through penalizing the prediction divergence between them.

Visualization. We further visualize the learnt representations of the baseline model and our model on CIFAR-10 with 1000 labels. We use the first essential networks in 'Baseline' and 'Our Model' for comparison. Figure 9 shows the features of the last hidden layer projected to 2 dimensions by using t-SNE [23]. The instances are all from the test data, and different classes are encoded by different

colors. It can be observed that the learned representations of the proposed model are more concentrated, and can be easily divided into different groups.

4.5. Results on CIFAR-100

CIFAR-100 is a more challenging benchmark for semi-supervised classification due to the reason that there are 100 categories. There are a few methods tested on this benchmark. Table 4 shows the results of our model and the competing methods. Similar to the results achieved on the other benchmarks, 'Our Model' significantly improves 'Baseline' in both cases. When given 10000 labels, the test error rate is reduced to 35.28%, which is lower than the previous state-of-the-art result (37.97%). The results on CIFAR-100 verify the effectiveness of our enhanced mutual learning when dealing with more difficult benchmarks.

5. Conclusion

This work explores how to enhance mutual learning between deep convolutional networks for improving semi-supervised classification. We show that simply minimizing the prediction divergence between two separate essential networks may not fully leverage the difference between them. To capture this information, we propose a complementary correction network, built on top of the essential networks, to correct the prediction of one network, conditioned on the features learnt by another. The resulting more accurate class predictions for the unlabeled instances are used as the training targets to make the second essential network become more complementary to the first one. As a result, our enhanced mutual learning model leads to significant performance gains, due to the reason that the learnt knowledge can be ultimately transferred to the first essential network. Our experiments demonstrate that the proposed approach improves the state-of-the-art results on multiple semi-supervised classification benchmarks.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *Proc. International Conference on Machine Learning*, 2017.

[2] P. Bachman, O. AIsharif, and D. Precup. Learning with pseudo-ensembles. In *Proc. Advances in Neural Information Processing Systmes*, pages 3365 – 3373, 2014.

[3] T. Batra and D. Parikh. Cooperative learning with visual attributes. In *arXiv preprint arXiv:1705.05512*, 2017.

[4] Z. Dai, Z. Yang, F. Yang, W. Cohen, and R. Salakhutdinov. Good semi-supervised learning that requires a bad GAN. In *Proc. Advances in Neural Information Processing Systems*, pages 6513 – 6523, 2017.

[5] Z. Ding, N. Nasrabadi, and Y. Fu. Semi-supervised deep domain adaptation via coupled neural networks. *IEEE Transactions on Image Processing*, 27(11):5214 – 5224, 2018.

[6] V. Dumoulin, I. Belghazi, B. Poole, O. Mastropietro, A. Lamb, M. Arjovsky, and A. Courville. Adversarially learned inference. In *Proc. International Conference on Learning Representation*, 2017.

[7] Z. Gan, L. Chen, W. Wang, Y. Pu, Y. Zhang, H. Liu, C. Li, and L. Carin. Triangle generative adversarial networks. In *Proc. Advances in Neural Information Processing Systems*, 2017.

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Proc. Advances in Neural Information Processing Systems*, pages 2672 – 2680, 2014.

[9] I. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *Proc. International Conference on Learning Representation*, 2015.

[10] Y. Grandvalet and Y. Bengio. Semi-supervised learning by entropy minimization. In *Proc. Advances in Neural Information Processing Systems*, 2004.

[11] D. He, Y. Xia, T. Qin, L. Wang, N. Yu, T. Liu, and W. Ma. Dual learning for machine translation. In *Proc. Advances in Neural Information Processing Systems*, pages 820 – 828, 2016.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pages 770 – 778, 2016.

[13] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. In *Proc. NIPS Deep Learning and Representation Learning Workshop*, 2014.

[14] D. Kingma, S. Mohamed, D. Rezende, and M. Welling. Semi-supervised learning with deep generative models. In *Proc. Neural Information Processing Systmes*, pages 3581 – 3589, 2017.

[15] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. In *Univ. Toronto, Toronto, ON, Canada, Tech. Rep.*, 2009.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. Neural Information Processing Systmes*, pages 1106 – 1114, 2014.

[17] A. Kumar, P. Sattigeri, and T. Fletcher. Semi-supervised learning with GANs: manifold invariance with improve inference. In *Proc. Advances in Neural Information Processing Systmes*, pages 5534 – 5544, 2017.

[18] S. Laine and T. Aila. Temporal ensembling for semi-supervised learning. In *Proc. International Conference on Learning Representations*, 2017.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278 – 2324, 1998.

[20] C. Li, K. Xu, J. Zhu, and B. Zhang. Triple generative adversarial nets. In *Proc. Advances in Neural Information Processing Systems*, pages 1195 – 1204, 2017.

[21] C. Li, J. zhu, and B. Zhang. Max-margin deep generative models for (semi-) supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[22] Y. Luo, J. Zhu, M. Li, Y. Ren, and B. Zhang. Smooth neighbors on teacher graphs for semi-supervised learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[23] L. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(11):2579 – 2605, 2008.

[24] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2018.

[25] T. Miyato, S. Maeda, S. Ishii, and M. Koyama. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[26] T. Miyato, S. Maeda, M. Koyama, K. Nakae, and S. Ishii. Distributional smoothing with virtual adversarial training. In *Proc. International Conference on Learning Representations*, 2016.

[27] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Ng. Reading digits in natural images with unsupervised feature learning. In *Proc. NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[28] A. Odena. Semi-supervised learning with generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2016.

[29] S. Park, J. Park, S. Shin, and I. Moon. Adversarial dropout for supervised and semi-supervised learning. In *Proc. AAAI Conference on Artificial Intelligence*, 2018.

[30] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. International Conference on Learning Representation*, 2016.

[31] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko. Semi-supervised learning with ladder networks. In *Proc. Neural Information Processing Systmes*, pages 3546 – 3554, 2015.

[32] M. Sajjadi, M. Javanmardi, and T. Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Proc. Advances in Neural Information Processing Systems*, pages 1163 – 1171, 2016.

[33] T. Salimans, I. Goodfellow, W. Zaremba, and V. Cheung. Improved techniques for training GANs. In *Proc. Neural Information Processing Systmes*, pages 2234 – 2242, 2016.

[34] U. Shaham, K. Stanton, H. Li, B. Nadler, R. Basri, and Y. Kluger. SpectralNet: spectral clustering using deep neural networks. In *Proc. International Conference on Learning Representation*, 2018.

[35] R. Shu, H. Bui, H. Narui, and S. Ermon. A DIRT-T approach to unsupervised domain adaptation. In *Proc. International Conference on Learning Representations*, 2018.

[36] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. International Conference on Learning Representation*, 2015.

[37] J. Springenberg. Unsupervised and semi-supervised learning with categorical generative adversarial networks. In *Proc. International Conference on Learning Representations*, 2016.

[38] A. Tarvainen and H. Valpola. Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In *Proc. Advances in Neural Information Processing Systems*, 2017.

[39] X. Wei, B. Gong, Z. Liu, W. Lu, and L. Wang. Improving the improved training of wasserstein GANs: a consistency term and its dual effect. In *Proc. International Conference on Learning Representations*, 2018.

[40] H. Wu and S. Prasad. Semi-supervised deep learning using pseudo labels for hyperspectral image classification. *IEEE Transactions on Image Processing*, 27(3):1259 – 1270, 2018.

[41] S. Wu, Q. Ji, S. Wang, H. Wong, Z. Yu, and Y. Xu. Semi-supervised image classification with self-paced cross-task networks. *IEEE Transactions on Multimedia*, 20(4):851–865, 2018.

[42] Y. Zhang, T. Xiang, T. Hospedales, and H. Lu. Deep mutual learning. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2018.