TransMatch: A Transfer-Learning Scheme for Semi-Supervised Few-Shot Learning

Zhongjie Yu*[1], Lin Chen†[2], Zhongwei Cheng[2], and Jiebo Luo[3]

[1]University of Wisconsin-Madison
[2]Futurewei Technologies
[3]University of Rochester

Abstract

The successful application of deep learning to many visual recognition tasks relies heavily on the availability of a large amount of labeled data which is usually expensive to obtain. The few-shot learning problem has attracted increasing attention from researchers for building a robust model upon only a few labeled samples. Most existing works tackle this problem under the meta-learning framework by mimicking the few-shot learning task with an episodic training strategy. In this paper, we propose a new transfer-learning framework for semi-supervised few-shot learning to fully utilize the auxiliary information from labeled base-class data and unlabeled novel-class data. The framework consists of three components: 1) pre-training a feature extractor on base-class data; 2) using the feature extractor to initialize the classifier weights for the novel classes; and 3) further updating the model with a semi-supervised learning method. Under the proposed framework, we develop a novel method for semi-supervised few-shot learning called TransMatch *by instantiating the three components with Imprinting and MixMatch. Extensive experiments on two popular benchmark datasets for few-shot learning, CUB-200-2011 and miniImageNet, demonstrate that our proposed method can effectively utilize the auxiliary information from labeled base-class data and unlabeled novel-class data to significantly improve the accuracy of few-shot learning task.*

1. Introduction

Deep learning methods have been making impressive progress in different areas of artificial intelligence in recent years. Nevertheless, most of the popular deep learning methods require a large amount of labeled data which is usually very expensive and time-consuming to collect. The straightforward adoption of deep learning methods with a limited amount of labeled data usually leads to overfitting. Therefore, the question of whether it is able to learn a robust model from only a limited amount of labeled data arises. It is well-known that humans have the ability to learn from a single or very few labeled samples. This motivates recent research efforts on learning a novel concept from a single or a few examples, *i.e.*, few-shot learning.

In the past couple of years, an increasing number of few-shot learning methods have been proposed. One family of work focuses on training the model under the *meta-learning* framework based on an episodic training strategy [25]. In particular, a sequence of episodes are randomly sampled where each episode consists of a few samples in the base classes to mimic the test scenario where only a few labeled samples of the novel classes are available. The labeled samples in each episode are divided into supports and queries, where supports are used for building the classifier and queries are used for evaluating. At the same time, another family of work focuses on how to learn a classifier for the novel classes with only few-shot examples by transferring the knowledge from a model pre-trained on large amount of data from the base classes [16, 17]. This paradigm shares similarity with human behaviors, by transferring past experience to new tasks. We denote this family of methods as *transfer-learning* based methods. Our method is inspired by the latter family of work and aims to learn a good classifier for the novel classes of few-shot examples with the help of the pre-trained classifier on abundant data from base classes and auxiliary unlabeled data from novel classes.

We believe the sufficient and proper utilization of extra information is crucial to the success of applying few-shot

*This work was done during Zhongjie's internship at Futurewei Technologies.

†Corresponding author: Lin Chen. Email: gggchenlin@gmail.com

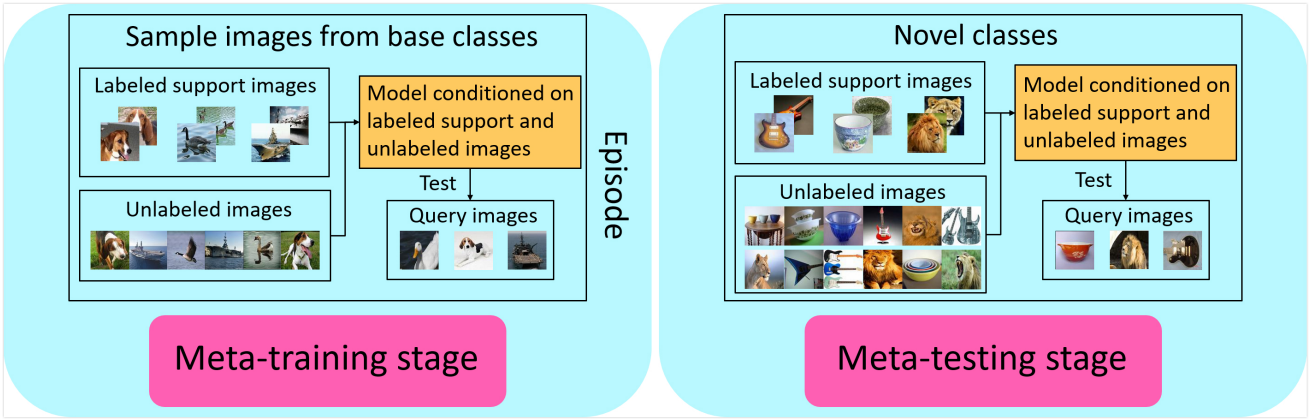

Figure 1. An overview of meta-learning based semi-supervised few-shot classification framework. Unlabeled images are required during training to allow the meta-learner learn how to leverage unlabeled images for classification.

learning. Such extra information can exist in various forms, while in this work, we focus on leveraging extra information from the labeled base-class data and unlabeled novel-class data. These two types of information are usually easy to obtain. Many existing large-scale datasets for visual recognition tasks can be used for pre-training a model which can be later transferred to a new task. Meanwhile, it is also relatively easy to acquire a large amount of unlabeled data for a new task. Thus, a new paradigm called semi-supervised few-shot learning arises recently.

A representative work for semi-supervised few-shot learning [19] employed the meta-learning framework and enhanced the prototypical networks [22] to use unlabeled data. In each episode during meta-training, the unlabeled data for base classes was included to mimic the test scenario where the unlabeled data for novel classes would be available. Liu *et al.* [11] proposed transductive propagation to incorporate the popular label propagation method to utilize the unlabeled data in episodic training. These works demonstrated that considering the unlabeled data helped to improve the accuracy of few-shot classification under the meta-learning framework.

In this paper, we propose a new framework for semi-supervised few-shot learning to fully utilize the auxiliary information from labeled base-class data and unlabeled novel-class data. The flowchart of our proposed framework is shown in Fig. 2, which consists of three components. We first train a model using the large amount of labeled data from the base classes, encoding the knowledge from base-class data into the pre-trained model. Then this pre-trained model is adopted as a feature extractor to generate the feature embeddings of the labeled few-shot examples from the novel classes, which can be directly used to imprint classifier weights for the novel classes or as the initialization of classifier weights for further fine-tuning, following the transfer-learning framework [16]. Different from meta-learning, unlabeled images are no longer needed during pre-training on base classes, and could be directly utilized upon this imprinted classifier with state-of-the-art semi-supervised method such as MixMatch [1]. To the best of our knowledge, this is the first work of semi-supervised few-shot learning under the transfer-learning framework in contrast to the meta-learning framework.

In summary, the contributions of our work are:

1. We propose a new transfer-learning framework for semi-supervised few-shot learning, which can fully utilize the auxiliary information from labeled base-class data and unlabeled novel-class data.

2. We develop a new method called *TransMatch* under the proposed framework. TransMatch integrates the advantages of transfer-learning based few-shot learning methods and semi-supervised learning methods, and is different from the previous work on meta-learning based methods.

3. We conduct extensive experiments on two popular benchmark datasets for few-shot learning to demonstrate that our method can effectively leverage unlabeled data in few-shot learning and achieve new state-of-the-art results.

2. Related Work

In this section, we review the related work to our proposed transfer-learning based semi-supervised few-shot learning framework.

2.1. Few-Shot Learning

Few-shot learning has attracted increasing attention in recent years due to the high cost of collecting labeled data. Existing work can be roughly categorized into (i) meta-learning methods, and (ii) transfer-learning methods.

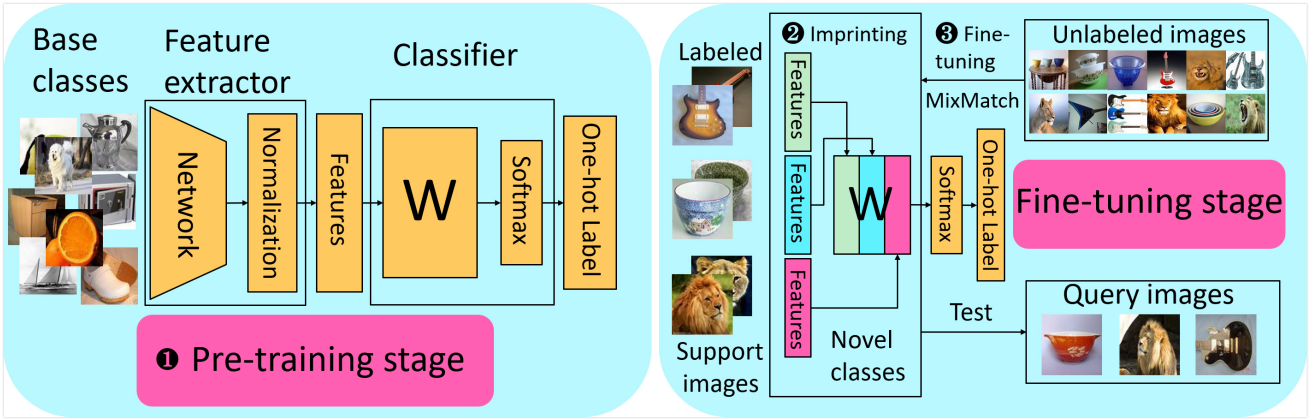

Figure 2. Our proposed framework of transfer-learning scheme for semi-supervised few-shot learning. We first pre-train a classifier from base-class images. Then use it as a feature extractor to initialize the weights for novel-class classifier. Finally, we further fine-tune the novel-class classifier with unlabeled images by semi-supervised learning method MixMatch.

Meta-learning based method: Meta-learning based few-shot learning, also known as *learning to learn*, aims to learn a paradigm that can be adapted to recognize novel classes with only few-shot training examples. Meta-learning based methods usually consist of two stages: 1) meta-training; and 2) meta-testing. In the meta-training stage, a sequence of episodes are randomly sampled from the examples of base classes where each episode contains K support examples and Q query examples from N classes, denoted as an N-way K-shot episode. In this way, the meta-training stage can mimic the few-shot testing stage where only a few examples per class are available. The meta-learning based methods can be further divided into two categories: a) metric-based methods; and b) optimization-based methods.

a) Metric-based methods have been proposed in many existing work [9, 15, 22, 23, 25]. These methods mainly focus on learning a good metric to measure the distance or similarity among support images and query images. For example, prototypical networks [22] calculated the distance of the prototype representations of each class between supports and queries. Relation Net [23] implemented a network to measure the relation similarities between the supports and queries. Nearest Neighbour Neural Network [9] explored the nearest neighbors in local descriptors of feature embeddings.

b) Optimization-based methods aim to design an optimization algorithm that can adapt the information during meta-training stage to the meta-testing stage. Meta-LSTM [18] formulated the problem as an LSTM-based meta-learning algorithm to update the optimization algorithm in few-shot learning. MAML [4] learned an optimization method that can follow the fast gradient direction to rapidly learn the classifier for novel classes. LEO [20] decoupled the gradient-based adaptation process with high-dimensional parameters to few-shot scenarios.

Transfer-learning based methods: Transfer-learning based methods are different from meta-learning based methods, as they do not use the episodic training strategy. Instead, such methods can use conventional techniques to pre-train a model on the large amount of data from the base classes. The pre-trained model is then adapted to the few-shot learning task of recognizing novel classes. Qi *et al.* [16] proposed to imprint the classifier weights of novel classes by the mean vectors of the feature embeddings of few-shot examples. Qiao *et al.* [17] learned a mapping function from the activations (*i.e.*, feature embeddings) of novel class examples to classifier weights. Gidaris *et al.* [5] proposed an attention module to dynamically predict the classifier weights for novel classes. Chen *et al.* [2] shown such transfer-learning based methods can achieve competitive performance as meta-learning based methods. Our proposed framework shares a similar idea with [16] by pre-training a feature extractor and uses it to extract features for few-shot examples from novel classes which are used to imprint classifiers weights.

2.2. Semi-Supervised Learning

Semi-supervised learning focuses on developing algorithms to learn from unlabeled and labeled data. Existing work can be roughly categorized into (i) consistency regularization methods, and (ii) entropy minimization methods.

Consistency regularization methods: Consistency regularization methods mainly focus on adding noise and augmentation to images without changing their label distribution. Π-Model [7] added a loss term to regularize the model by stochastic augmentation. Mean Teacher [24] improved Π-Model by using the exponential moving average of parameters. Virtual Adversarial Training (VAT) [13] regularized the model by adding local perturbation on unlabeled

data.

Entropy minimization methods: This family of methods focuses on giving low entropy for unlabeled data. It is initially proposed by [6] which minimized conditional entropy of unlabeled data. Pseudo-Label [8] minimized the entropy directly by predicting the labels for unlabeled data and used this in cross-entropy, showing its good performance.

MixMatch [1] united different kinds of consistency regularization and entropy minimization methods and achieved state-of-the-art performance by a large margin comparing with all the previous methods. It is a holistic method in semi-supervised learning and we would introduce briefly in Section 3.3. Due to its good performance, we adopt MixMatch in our framework, and we also compared with using other mainstream semi-supervised learning methods in the experiments. Semi-supervised learning methods are usually compared on small datasets [1, 13, 14] where there is a small amount of labeled data. But the number of labeled images in typical semi-supervised learning is still greater than few-shot learning. The techniques for semi-supervised method may not be directly used for few-shot setting, which is also demonstrated in our experiments that naively applying MixMatch to few-shot learning may lead to poor performance especially in 1-shot and 2-shot.

2.3. Semi-Supervised Few-Shot Learning

When there are few-shot examples for novel classes, it is straightforward to utilize extra unlabeled data to improve the learning. This leads to the family of semi-supervised few-shot learning methods (SSFSL). There are very few works in this direction. Ren *et al.* [19] extended prototypical networks to incorporate unlabeled data by producing prototypes for the unlabeled data. Liu *et al.* [11] constructed a graph between labeled and unlabeled data and utilize label propagation to obtain the labels of unlabeled data. Sun *et al.* [10] applied self-training by adding the confident prediction of unlabeled to the labeled training set in each round of optimization.

However, all existing semi-supervised few-shot learning methods are meta-learning based methods as in Fig. 1. As shown in [2], transfer-learning based method can achieve competitive performance compared with meta-learning based methods. This motivates our work. We need to emphasize that meta-learning based methods have shown their success to utilize unlabeled data by integrating unlabeled data in episodic training. However, this episodic training strategy is different from typical semi-supervised learning and it is not appropriate to combine them together directly. The techniques of leveraging unlabeled data in existing SSFSL methods are not state-of-the-art in semi-supervised areas and the more powerful and holistic methods like MixMatch would be difficult to integrate in meta-learning framework. Meanwhile, directly applying semi-supervised methods to utilize unlabeled data during test may lead to bad performance due to the extreme small number of labeled data.

3. The Proposed Framework

In this section, we introduce our proposed transfer-learning framework for semi-supervised few-shot learning. The flowchart is illustrated in Fig. 2, which contains three modules: 1) pre-training a feature extractor on base-class data; 2) use the feature extractor to extract features from novel-class data and imprint novel-class classifier weights; and 3) further fine-tuning the model by semi-supervised learning method. Before elaborating the details of each module, let us first introduce our problem definition.

Problem definition: We have a large-scale dataset $\mathcal{D}_{base}$ containing many-shot labeled examples from each base class in $\mathcal{C}_{base}$ and a small-scale dataset $\mathcal{D}_{novel}$ of only few-shot labeled examples and many-shot unlabeled examples from each novel class in $\mathcal{C}_{novel}$, where $\mathcal{C}_{novel}$ is disjoint from $\mathcal{C}_{base}$. The task of semi-supervised few-shot learning is to learn a robust classifier using both the few-shot labeled examples and many-shot unlabeled examples in $\mathcal{D}_{novel}$ with the examples in $\mathcal{D}_{base}$ as auxiliary data. Usually in a conventional few-shot learning task, a small support set of N classes with K images per class is sampled from $\mathcal{D}_{novel}$, leading to the N-way-K-shot problem. In semi-supervised few-shot learning, additional U unlabeled images are sampled from each of the N novel classes or distractor classes.

3.1. Part I: Pre-train Feature Extractor

The first module of our framework, as shown in the left part of Fig. 2, is a pre-training module, which relies on the many-shot examples from base classes, $\mathcal{D}_{base}$, to train a base model which encodes as much as possible the information of $\mathcal{D}_{base}$ and can be used in the later stage of few-shot learning as prior information, similar to human intelligence. This is different from conventional meta-learning based few-shot learning as shown in Fig. 1, where an episodic training strategy is employed for base classes as well to mimic the few-shot scenario in the testing phase.

3.2. Part II: Classifier Weight Imprinting

The weight imprinting method was proposed by [16], and has achieved impressive performance in the few-shot learning task as a representative of transfer-learning based few-shot learning method. Specifically, it directly sets the classifier weights by the mean feature vectors of the N-way-K-shot examples, where features are obtained by the model from the pre-training stage. For convenience, we denote the classifier on large scale base classes as $f(\mathbf{x}) = f^{base}(f^e(\mathbf{x}))$, where $\mathbf{x}$ is an input example, $f^e(\cdot)$ is the

feature extractor and $f^{base}(\cdot)$ is the classifier. We have $f^e(\mathbf{x}) \in \mathcal{R}^d$ and $f^{base}(\cdot) \in \mathcal{R}^{|\mathcal{C}_{base}|}$.

Given the N-way-K-shot examples from novel classes and let us denote them as $\mathcal{D}_{novel} = \{\mathbf{x}_k^c|_{k=1...K,\ c=1...N}\}$ with $\mathbf{x}_k^c$ as the k-th example in c-th class. We can use the feature extractor learned on base classes to extract features for the N-way-K-shot examples, denoted as $f^e(\mathbf{x}_k^c)$. Meanwhile, let us write the classifier for novel classes as $f^{novel}(\mathbf{x}) = \mathbf{W}'\mathbf{x}$, where $\mathbf{W} = [\mathbf{w}_1, ..., \mathbf{w}_N] \in \mathcal{R}^{d\times N}$. Note that we omit the bias for simplicity. By normalizing the weight $\mathbf{w}_c$ and the feature vector $\mathbf{x}$ onto a unit ball, the aforementioned equation can be further simplified as

$$f^{novel}(\mathbf{x}) = [\cos(\theta(\mathbf{w}_1, \mathbf{x}), ..., \cos(\theta(\mathbf{w}_N, \mathbf{x}))]', \quad (1)$$

where $\theta(\mathbf{w}_i, \mathbf{x})$ denotes the angle between $\mathbf{w}_i$ and $\mathbf{x}$, and the classification for a given example $\mathbf{x}$ is based on computing the cosine similarity between every $\mathbf{w}_k$ and $\mathbf{x}$, and predict the label of $\mathbf{x}$ based on maximum similarity score.

In this sense, there is a duality between $\mathbf{w}_i$ and $\mathbf{x}$. Based on this observation, weight imprinting uses the mean feature vectors of the few-shot examples to imprint $\mathbf{w}_c$, *i.e.*, by setting

$$\mathbf{w}_c = \frac{1}{K}\sum_{k=1}^{K} f^e(\mathbf{x}_k^c). \quad (2)$$

The classification of an given example $\mathbf{x}$ can be also deemed as computing the mean of the similarities between $\mathbf{x}$ and all K-shot examples.

By imprinting the classifier weights with mean feature vectors of the few-shot examples, it provides a better initialization of classifier weights to reduce the intra-class variations of features and benefits fine-tuning the new classifier for novel classes. Experimental results show that it can achieve good performance even without fine-tuning.

3.3. Part III: Semi-Supervised Fine-tuning

After we get the classifier which fully absorbs the information from base classes with a better initialization by imprinting, we fine-tune this classifier during test when there is unlabeled data. This fine-tuning process is the same as semi-supervised training. Any semi-supervised learning can be applied, and in this work we employed MixMatch [1] not only because of its excellent performance in the semi-supervised learning task, but also because it is a holistic method to leverage unlabeled data in semi-supervised learning area.

MixMatch combines multiple existing improvements from state-of-the-art semi-supervised learning methods which is discussed in Section 2.2. In our setting, we denote $\mathcal{L} = \{(\mathbf{x}_i, p_i)\}_{i=1}^B$ as a mini-batch of B labeled examples with p_i as the label, and $\mathcal{U} = \{\mathbf{x}_u\}_{u=1}^U$ as a mini-batch of U unlabeled examples. The imprinted classifier from Part II can be used to obtain estimated labels for the examples in $\mathcal{U}$, *i.e.*, $f^{novel}(\mathbf{x}_u)$. We will omit the superscript ^{novel} for the ease of illustration when there is no confusion. For robustness, we augment each example M times to get M versions of each unlabeled data, *i.e.*, $\{\mathbf{x}_{u,1}, ..., \mathbf{x}_{u,M}\}$, and use the mean prediction as the label estimation: $\bar{p}_u = \frac{1}{M}\sum_{i=1}^M f(\mathbf{x}_{u,i})$. The sharpen operation is used to enhance to prediction as $p_u = \bar{p}_u^{\frac{1}{T}} / \sum_{j=1}^N (\bar{p}_u)_j^{\frac{1}{T}}$, we set $T = 0.5$ in the experiments. The same data augmentation is also applied to labeled examples in $\mathcal{L}$. Following [1], we concatenate $\mathcal{L}$ and $\mathcal{U}$ and shuffle the examples, *i.e.*, $\mathcal{W} = \text{Shuffle}(\text{Concat}(\mathcal{L}, \mathcal{U}))$, and then split this set into two new sets:

$$\begin{aligned}\mathcal{X}_1' &= \{\text{MixUp}(\mathcal{L}_i, \mathcal{W}_i) \quad |i \in 1, \ldots, |\mathcal{L}|\}, \\ \mathcal{X}_2' &= \left\{\text{MixUp}\left(\mathcal{U}_i, \mathcal{W}_{i+|\mathcal{L}|}\right) \ |i \in 1, \ldots, |\mathcal{U}|\right\},\end{aligned}$$

where MixUp is defined as

$$\begin{aligned}&\text{MixUp}((\mathbf{x}_1, p_1), (\mathbf{x}_2, p_2)) \\ &\quad = ((\lambda'\mathbf{x}_1 + (1-\lambda')\mathbf{x}_2), (\lambda' p_1 + (1-\lambda')p_2)) \quad (3)\end{aligned}$$

with $\lambda' = \max(\lambda, 1-\lambda)$. The parameter λ is randomly generated from a beta distribution $\text{Beta}(\alpha, \alpha)$. The objective function to minimize is defined as

$$\ell = \ell_1 + \gamma\ell_2, \quad (4)$$

where

$$\ell_1 = -\frac{1}{|\mathcal{X}_1'|}\sum_{(\mathbf{x},p)\in\mathcal{X}_1'} p\log(f(\mathbf{x})), \quad (5)$$

is cross-entropy loss, and

$$\ell_2 = \frac{1}{N|\mathcal{X}_2'|}\sum_{(\mathbf{x},p)\in\mathcal{X}_2'} \|p - f(\mathbf{x})\|_2^2. \quad (6)$$

is consistency regularization loss in [21]. The details of our algorithm is summarized in Algorithm 1.

Algorithm 1 Algorithm for our proposed TransMatch

Input: An auxiliary dataset $\mathcal{D}_{base}$ with examples from $\mathcal{C}_{base}$ (base classes), N-way-K-shot dataset $\mathcal{D}_l = \{\mathbf{x}_{nk}, p|n = 1, \cdots, N; k = 1, \cdots, K\}$ with $p \in \mathcal{C}_{novel}$ (novel classes), and $\mathcal{D}_u = \{\mathbf{x}_u|u = 1, \cdots, U\}$

Output: N-way-K-shot classifier f^{novel} for $\mathcal{D}_l$

1: Pre-train a base network on all examples in $\mathcal{D}_{base}$ and denote it as $f^{base}(f^e(\mathbf{x}))$;
2: Apply the feature extractor $f^e(\mathbf{x})$ to extract features on $\mathcal{D}_l$, then use these features to imprint the weights of the novel classifier f^{novel};
3: Apply semi-supervised learning method, MixMatch, to update the novel classifier f^{novel} with both $\mathcal{D}_l$ and $\mathcal{D}_u$;

4. Experiments

In this section, we evaluate our proposed TransMatch and compare with state-of-the-art few-shot learning methods on two popular benchmark datasets for few-shot learning, including miniImageNet and CUB-200-2011.

4.1. Experiments on miniImageNet

Dataset configuration: The miniImageNet dataset was originally proposed by [25]. It has been widely used for evaluating few-shot learning methods. It consists of 60,000 color images from 100 classes with 600 examples per class, which is a simplified version of ILSVRC 2015 by [3]. We follow the split given by [18] consisting of 64 base classes, 16 validation classes and 20 novel classes. We randomly select K (*resp.* U) examples from each novel class as the few-shot labeled (*unlabeled*) examples, and Q images from the rest as the test examples. In the experiments, we set $N = 5$, $K = \{1, 5\}$, $Q = 15$ and study the effect of using different values of U. We repeat the test experiments 600 times and report the mean accuracy with the 95% confidence interval.

Compared methods: The miniImageNet dataset has been widely used for evaluating the performance of few-shot learning methods, and is a good benchmark to compare state-of-the-art methods. In particular, we compare with several conventional few-shot learning methods, as well as state-of-the-art semi-supervised few-shot learning methods including the semi-supervised extension to Prototypical Networks by [19] (Soft k-Means, Soft k-Means+Cluster, Masked Soft k-Means), and TPN-semi in [11]. We also re-implement Soft k-Means, Soft k-Means+Cluster, Masked Soft k-Means with the same backbone (*i.e.*, WRN-28-10) as our method for fair comparison. As the area of semi-supervised few-shot learning has not been explored much yet, we also conduct extensive experiments to evaluate the performance of utilizing unlabeled data by our TransMatch under different few-shot settings.

Implementation details: Following the work [17] for transfer-learning based method on miniImageNet, we use the wide residual network (*i.e.*, WRN-28-10) [27] as the backbone for our base model f^{base}. We train it from scratch using the examples from the base classes. In particular, we first train a WRN-28-10 classification network on all examples from the 80 base and validation classes. We then replace the last layer of this network by a 256-d fully connected layer, followed by a L2 normalization layer and a 80-d classifier. We set the batch size to 128, and set learning-rate to 0.01 for the last two layers and 0.001 for all other layers. We reduce the learning rate by 0.1 every 10 epochs and train for a total of 28 epochs.

The base classifier f^{base} is used as the feature extractor to generate feature vectors for the few-shot examples from novel classes. We use the few-shot labeled examples to fine-tune the base classifier to novel classes. We also augment each labeled image for 10 times by random transformation and use the mean features to imprint the weights for novel classifier. We use a batch size of 16, and set 64 batches as an epoch[1]. We set weight decay to 0.04, learning rate to 0.001, and use SGD optimizer with a momentum of 0.9. For the fine-tuning stage, we set the parameters of MixMatch as follows. We set M (the times for augmentation) to 2, T (the temperature for the label distribution) to 0.5, γ (the weight for regularization term) to 5, α (the parameter in Beta distribution) to 0.75. Meanwhile we use an exponential moving average for model parameters when guessing labels. For 5-way-1-shot scenario, we fine-tune for 10 epochs when there are 20 or 50 unlabeled images, and 20 epochs when there are 100 or 200 unlabeled images. For 5-way-5-shot scenario, we fine-tune for 20 epochs when there are 20 and 50 unlabeled images, and 25 epochs when there are 100 and 200 unlabeled images. All the test results are based on 600 random experiments.

Results on miniImageNet: The results are summarized in Table 1. It is not surprising that our method outperforms conventional few-shot learning methods without using unlabeled by a large margin, as shown in the top portion of Table 1. Our method also outperforms state-of-the-art semi-supervised few-shot learning methods, which can be observed from the middle portion of Table 1. These results clearly show the superiority of our TransMatch as its effective utilization of information from unlabeled data.

Influence of unlabeled examples: In Table 2, we report the results using different numbers of unlabeled images. Note that Imprinting+FT stands for fine-tuning the imprinted classifier without unlabeled data. It is obvious that our TransMatch could achieve better performance with more unlabeled images. We also observe that the results begin to saturate after 100 unlabeled images for 1-shot setting. In general, the results show that our TransMatch can effecively utilize the unlabeled data.

Ablation study: We conduct an ablation study of our method without Imprinting or MixMatch. Without Imprinting, our method reduces to semi-supervised learning method, *i.e.*, MixMatch (Note here the feature extractor is still already trained from base classes) and without MixMatch, our method reduces to Imprinting. The results are shown in Fig. 3. It is clear that both MixMatch and Imprinting are worse than our TransMatch. The inferior performance of MixMatch to our TransMatch clearly shows that directly applying MixMatch to the few-shot setting cannot lead to good performance especially in 1-shot and 2-shot setting. This is due to the lack of labeled data, which makes it hard to fine-tune the classifier during test when there is unlabeled data. However, our proposed TransMatch can obtain a good initialization by incorporating weight imprinting

[1]We duplicate the labeled images dataset to make it larger, so that each batch may contain the same image multiple times.

Method		Type	1-shot	5-shot
Prototypical Net	[22]	Meta, Metric	49.42±0.78	68.20±0.66
TADAM	[15]	Meta, Metric	58.50±0.30	76.70±0.30
MAML	[4]	Meta, Optimization	48.70±1.84	63.11±0.92
SNAIL	[12]	Meta, Optimization	55.71±0.99	68.88±0.92
Activation Net	[17]	Transfer-learning	59.60±0.41	73.74±0.19
Imprinting	[16]	Transfer-learning	58.68±0.81	76.06±0.59
Soft k-Means	[19]	Semi, Meta-learning	50.09±0.45	64.59±0.28
Soft k-Means+Cluster	[19]	Semi, Meta-learning	49.03±0.24	63.08±0.18
Masked Soft k-Means	[19]	Semi, Meta-learning	50.41±0.31	64.39±0.24
TPN-semi	[11]	Semi, Meta-learning	52.78±0.27	66.42±0.21
Soft k-Means	(Re-implement with WRN-28-10)	Semi, Meta-learning	51.88±0.93	67.31±0.70
Soft k-Means+Cluster	(Re-implement with WRN-28-10)	Semi, Meta-learning	50.47±0.86	64.14±0.65
Masked Soft k-Means	(Re-implement with WRN-28-10)	Semi, Meta-learning	52.35±0.89	67.67±0.65
TransMatch	(100 unlabeled images per class)	Semi, Transfer-learning	**63.02±1.07**	81.19±0.59
TransMatch	(200 unlabeled images per class)	Semi, Transfer-learning	62.93±1.11	**82.24±0.59**

Table 1. Accuracy (in %) on miniImageNet with 95% confidence interval. Best results are in bold.

Method	# unlabeled	1-shot	5-shot
Imprinting	—	58.68±0.81	76.06±0.59
Imprinting+FT	0	55.60±0.77	74.17±0.60
TransMatch	20	58.43±0.93	76.43±0.61
TransMatch	50	61.21±1.03	79.30±0.59
TransMatch	100	**63.02±1.07**	81.19±0.59
TransMatch	200	62.93±1.11	**82.24±0.59**

Table 2. Accuracy (in %) with different number of unlabeled images on miniImageNet. Best results are in bold.

# shot	Method	Accuracy	Gain
1-shot	w/ Pseudo-Label	57.01 ± 1.13	+6.01
	w/ MixMatch	63.02 ± 1.07	
2-shot	w/ Pseudo-Label	70.07 ± 0.96	+2.29
	w/ MixMatch	72.36 ± 0.88	
3-shot	w/ Pseudo-Label	76.01 ± 0.81	+1.40
	w/ MixMatch	77.41 ± 0.76	
4-shot	w/ Pseudo-Label	78.35 ± 0.73	+1.39
	w/ MixMatch	79.74 ± 0.65	
5-shot	w/ Pseudo-Label	80.00 ± 0.66	+1.19
	w/ MixMatch	81.19 ± 0.59	

Table 3. Comparison of our method using different semi-supervised learning methods (*i.e.*, Pseudo-Label and MixMatch) in our framework both with 100 unlabeled images for 5-way classification on miniImageNet.

module.

We also observe a larger gain by our TransMatch over MixMatch when using a smaller number of shots. The gain shown in Fig. 3 is $\{11.02, 4.28, 2.92, 1.73, 1.22\}$ in {1, 2, 3, 4, 5}-shot setting. This is reasonable and worth attention as fewer shots means fewer labeled examples, which makes fine-tuning more difficult. Therefore, the importance of weight imprinting to give the classifier good initial weights

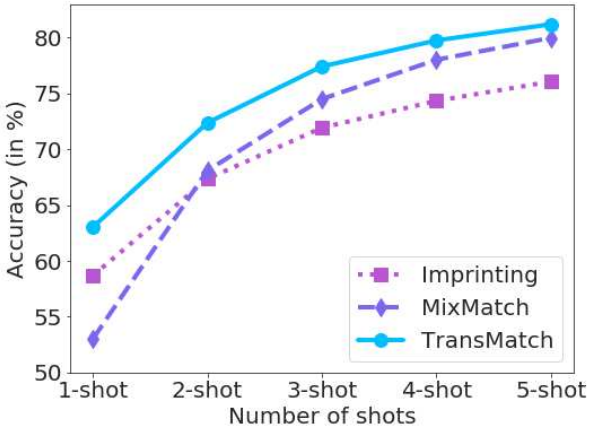

Figure 3. Comparison of Imprinting, MixMatch and our TransMatch both with 100 unlabeled images for 5-way classification with different number of shots on miniImageNet.

becomes more evident.

Comparing different semi-supervised learning methods: In addition to MixMatch [1], in this section, we also compare with other semi-supervised learning methods (*i.e.*, Pseudo-Label [8]) in order to understand the influence the semi-supervised learning module. The results, shown in Table 3, are consistent with our observations when using MixMatch as semi-supervised learning module. Since Pseudo-Label is worse than MixMatch, the overall performance of our method using Pseudo-Label is also worse than using MixMatch.

Influence of distractor classes: In typical semi-supervised learning, unlabeled images come from the same classes for the labeled images. This may not reflect realistic situations in real-world application. So we also study the influence

Distractor	Method	1-shot	5-shot
—	Imprinting	58.68 ± 0.81	76.06 ± 0.59
1-class	MixMatch	50.14 ± 1.06	79.32 ± 0.63
	TransMatch	$\mathbf{62.32 \pm 1.04}$	$\mathbf{80.28 \pm 0.62}$
2-class	MixMatch	50.68 ± 1.15	78.07 ± 0.69
	TransMatch	$\mathbf{60.41 \pm 1.02}$	$\mathbf{79.48 \pm 0.64}$
3-class	MixMatch	49.48 ± 1.16	77.48 ± 0.66
	TransMatch	$\mathbf{59.32 \pm 1.10}$	$\mathbf{79.29 \pm 0.62}$

Table 4. Accuracy (in %) of MixMatch and our TransMatch with 100 unlabeled images from $\{1, 2, 3\}$ *distractor* classes on miniImageNet. Note that Imprinting does not use any unlabeled image.

of distractor classes, and report the results of Imprinting, MixMatch, and our TransMatch when there are unlabeled images from various distractor classes. In our experiments, distractor classes are randomly chosen from the remaining classes which are disjoint with the novel classes during test. The results are shown in Table 4. We can observe that all the results for MixMatch degrade due to the distractor classes, while our TransMatch still outperforms Imprinting in all cases.

4.2. Experiments on CUB-200-2011

Dataset configuration: The CUB-200-2011 dataset (CUB) is originally proposed by [26] and contains 200 fine-grained classes of birds with 11,788 images in total (about 30 images per class for support images and 30 images per class for query images). We strictly follow the setup in [16] to ensure a fair comparison. In particular, we use the standard train/test split provided by the dataset, and treat the first 100 classes as the base classes $\mathcal{C}_{base}$ and the remaining 100 classes as the novel classes $\mathcal{C}_{novel}$. Therefore, we have $N = 100$. We use all the training examples from the base classes for large scale pre-training to obtain the base model f^{base} and use the few-shot examples from the novel classes to train f^{novel}. In the experiment, we set K to $\{1, 2, 5, 10, 20\}$ and use the rest images $\{29, 28, 25, 20, 10\}$ as unlabeled images for support images. All the remaining 30 images are still used for query images.

Implementation details: We are interested in performance of our TransMatch on the 100 novel classes, *i.e.*, the *transfer-learning* setting in [16]. In order to ensure fair comparison, we follow [16] and use Inception_v1 as our network backbone. We set the dimension of the fully connected embedding layer to 256, followed by an L2 normalization. We resize the input images to 256×256 and then randomly crop to 224×224. During the large scale pre-training stage, we set the initial learning rate to 0.001 and a $10\times$ multiplier for the embedding layer and classification layer. We reduce the learning rate by 0.1 after every 30 epochs, and train the model for a total of 90 epochs. During the fine-tuning stage, we set the number of batches to 64 for each epoch with a batch size of 64. By default, we set the weight decay to 0.0001, use a learning rate of 0.001, and train the model for 100 epochs. For the extreme case of 1-shot and 2-shot settings (100-way), we set the weight decay to 0.04, the learning rate to 0.0001 and early stopping at 10 epochs in order to avoid overfitting.

Model K=	1	2	5	10	20
Imprinting	26.08	34.13	43.34	48.91	52.94
Imprinting+FT	26.59	34.33	49.39	61.65	70.07
MixMatch	22.93	30.24	56.41	67.13	73.00
TransMatch	**28.02**	**38.05**	**59.83**	**68.60**	**74.61**

Table 5. Accuracy (in %) comparison on CUB-200-2011. Best results are in bold.

Model	# unlabeled	5-shot	10-shot
Imprinting [16]	—	43.34	48.91
Imprinting+FT [16]	0	49.39	61.65
TransMatch	5	52.90	63.79
TransMatch	10	54.78	66.21
TransMatch	15	56.86	67.71
TransMatch	20	59.25	68.60

Table 6. Accuracy (in %) comparison using different numbers of unlabeled images on CUB-200-2011.

Results on CUB-200-2011: We follow [16] to report the results of their proposed Imprinting, and Imprinting+FT. Then we evaluate the performance of our proposed TransMatch using different numbers of shots and unlabeled images. We compare TransMatch with Imprinting and MixMatch in Table 5, and the results show our proposed TransMatch achieves the best result which demonstrates its effectiveness in utilizing auxiliary labeled base-class data and unlabeled novel-class data. Table 6 shows the results of our TransMatch using different numbers of unlabeled images, and we can observe that better performance can be achieved with more unlabeled data. These results are similar to the results on miniImageNet dataset.

5. Conclusion

While almost all existing semi-supervised few-shot learning methods are based on the meta-learning framework, we propose a new transfer-learning framework for semi-supervised few-shot learning to effectively explore the information from labeled base-class data and unlabeled novel-class data. We develop a new method under the proposed framework by incorporating the state-of-the-art semi-supervised method MixMatch and few-shot learning method Imprinting, leading to a new method called TransMatch. Extensive experiments on two popular few-shot learning datasets show that our proposed TransMatch achieves the state-of-the-art results, which demonstrate its effectiveness in utilizing both the labeled base-class data and unlabeled novel-class data.

References

[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5050–5060, 2019.

[2] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Wang, and Jia-Bin Huang. A closer look at few-shot classification. In *ICLR*, 2019.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[4] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pages 1126–1135, 2017.

[5] Spyros Gidaris and Nikos Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, pages 4367–4375, 2018.

[6] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems*, pages 529–536, 2005.

[7] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *Proc. International Conference on Learning Representations (ICLR)*, 2017.

[8] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, 2013.

[9] Wenbin Li, Lei Wang, Jinglin Xu, Jing Huo, Yang Gao, and Jiebo Luo. Revisiting local descriptor based image-to-class measure for few-shot learning. In *CVPR*, pages 7260–7268, 2019.

[10] Xinzhe Li, Qianru Sun, Yaoyao Liu, Qin Zhou, Shibao Zheng, Tat-Seng Chua, and Bernt Schiele. Learning to self-train for semi-supervised few-shot classification. In *Advances in Neural Information Processing Systems*, pages 10276–10286, 2019.

[11] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. In *ICLR*, 2018.

[12] Nikhil Mishra, Mostafa Rohaninejad, Xi Chen, and Pieter Abbeel. A simple neural attentive meta-learner. In *ICLR*, 2018.

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

[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.

[15] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. TADAM: Task dependent adaptive metric for improved few-shot learning. In *Advances in Neural Information Processing Systems*, pages 721–731, 2018.

[16] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *CVPR*, pages 5822–5830, 2018.

[17] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan L Yuille. Few-shot image recognition by predicting parameters from activations. In *CVPR*, pages 7229–7238, 2018.

[18] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017.

[19] Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B. Tenenbaum, Hugo Larochelle, and Richard S. Zemel. Meta-learning for semi-supervised few-shot classification. In *ICLR*, 2018.

[20] Andrei A Rusu, Dushyant Rao, Jakub Sygnowski, Oriol Vinyals, Razvan Pascanu, Simon Osindero, and Raia Hadsell. Meta-learning with latent embedding optimization. In *ICLR*, 2019.

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

[22] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017.

[23] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *CVPR*, pages 1199–1208, 2018.

[24] 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.

[25] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016.

[26] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[27] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016.