

Learning to Learn Image Classifiers with Informative Visual Analogy

LINJUN ZHOU, Tsinghua University
PENG CUI, Tsinghua University
SHIQIANG YANG, Tsinghua University
WENWU ZHU, Tsinghua University
QI TIAN, University of Texas at San Antonio

In recent years, we witnessed a huge success of Convolutional Neural Networks on the task of the image classification. However, these models are notoriously data hungry and require tons of training images to learn the parameters. In contrast, people are far better learner who can learn a new concept very fast with only a few samples. The plausible mysteries making the difference are two fundamental learning mechanisms: learning to learn and learning by analogy. In this paper, we attempt to investigate a new human-like learning method by organically combining these two mechanisms. In particular, we study how to generalize the classification parameters of previously learned concepts to a new concept. we first propose a novel Visual Analogy Network Embedded Regression (VANER) model to jointly learn a low-dimensional embedding space and a linear mapping function from the embedding space to classification parameters for base classes. We then propose an out-of-sample embedding method to learn the embedding of a new class represented by a few samples through its visual analogy with base classes. By inputting the learned embedding into VANER, we can derive the classification parameters for the new class. These classification parameters are purely generalized from base classes (*i.e.* transferred classification parameters), while the samples in the new class, although only a few, can also be exploited to generate a set of classification parameters (*i.e.* model classification parameters). Therefore, we further investigate the fusion strategy of the two kinds of parameters so that the prior knowledge and data knowledge can be fully leveraged. We also conduct extensive experiments on ImageNet and the results show that our method can consistently and significantly outperform state-of-the-art baselines.

CCS Concepts: •**Computing methodologies** → **Object recognition**;

Additional Key Words and Phrases: Knowledge transfer, Low-shot image classification, Network embedding

ACM Reference format:

Linjun Zhou, Peng Cui, Shiqiang Yang, Wenwu Zhu, and Qi Tian. 2017. Learning to Learn Image Classifiers with Informative Visual Analogy. *ACM Trans. Web* 9, 4, Article 39 (September 2017), 15 pages.
DOI: 0000001.0000001

This work is supported by the National Science Foundation, under grant CNS-0435060, grant CCR-0325197 and grant EN-CS-0329609.

Authors' addresses: G. Zhou, Computer Science Department, College of William and Mary, 104 Jameson Rd, Williamsburg, PA 23185, US; V. Béranger, Inria Paris-Rocquencourt, Rocquencourt, France; A. Patel, Rajiv Gandhi University, Rono-Hills, Doimukh, Arunachal Pradesh, India; H. Chan, Tsinghua University, 30 Shuangqing Rd, Haidian Qu, Beijing Shi, China; T. Yan, Eaton Innovation Center, Prague, Czech Republic; T. He, C. Huang, J. A. Stankovic University of Virginia, School of Engineering Charlottesville, VA 22903, USA; T. F. Abdelzaker, (Current address) NASA Ames Research Center, Moffett Field, California 94035.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2017 Copyright held by the owner/author(s). Publication rights licensed to ACM. 1559-1131/2017/9-ART39 \$15.00
DOI: 0000001.0000001

1 INTRODUCTION

The recent progress of machine learning, especially the emergence of deep learning, has advanced the image classification performance into an unprecedented level. The error rates on large-scale benchmark datasets has been halved and halved again, even approaching human-level performance on some object recognition benchmarks. Despite the success, the state-of-the-art models are notoriously data hungry, requiring tons of samples for parameter learning. In real cases, however, the visual phenomena follows a long-tail distribution [23] where only a few sub-categories are data-rich and the rest are with limited training samples. How to learn a classifier from as fewer samples as possible is critical for real applications and fundamental for exploring new learning mechanisms.

Compared with machines, people are far better learners as they are capable of learning models from very limited samples of a new category and make accurate predictions and judgements accordingly. An intuitive example is that a baby learner can learn to recognize a wolf with only a few sample images provided that he/she has been able to successfully recognize a dog. The key mystery making the difference is that people have strong prior knowledge to generalize across different categories [9]. It means that people do not need to learn a new classifier (e.g. wolf) from scratch as most machine learning methods, but generalize and adapt the previously learned classifiers (e.g. dog) towards the new category. A major way to acquire these prior knowledge is through learning to learn from previous experience. In the image classification scenario, learning to learn refers to the mechanisms that learning to recognize a new concept can be accelerated by previous learning of other related concepts.

A typical image classifier is constituted by representation and classification steps, leading to two fundamental problems in learning to learn image classifiers: (1) how to generalize the representations from previous concepts to a new concept, and (2) how to generalize the classification parameters of previous concepts to a new concept. In literature, transfer learning and domain adaptation methods [10] are proposed with a similar notion, mainly focusing on the problem of representation generalization across different domains and tasks. With the development of CNN-based image classification models, the high-level representations learned from very large scale labeled dataset, e.g. the fc7 layer in AlexNet, are demonstrated to have good transfer ability across different concepts or even different datasets [19], which significantly alleviate the representation generalization problem. However, how to generalize the classification parameters in deep models (e.g. the fc7 layer in AlexNet) from well-trained concepts to a new concept (with only a few samples) is largely ignored by previous studies.

In this paper, we target the following problem. Given a well-trained N -class CNN model for N base classes, how to learn a binary classifier for the $(N + 1)^{th}$ class with only a few samples? More specifically, we constrain the setting to let $(N + 1)^{th}$ class share the same representation space as N base classes, i.e. we directly copy the representation layers of the N -class CNN model to the $(N + 1)^{th}$ class, which is a common way in deep representation transfer [8, 20, 21]. Such a setting provides a reasonable and fair foundation for investigating how to optimally generalize classification parameters. Given a new class, the key problem is to identify which base classes' classification parameters should be transferred.

Learning by analogy has been proved to be a fundamental building block in human learning process [3], and share similar context with our problem. When we face a new situation, we recall a similar situation by matching them up, and then we learn from it. Similarly, in the previous example of dog and wolf, we have a plausible explanation on the fast learning of wolf that a human learner selects dog from the base classes by visual analogy and transfers its classification

parameters for wolf classification. In this sense, visual analogy provides effective and informative clue for generalizing image classifiers in a way of human-like learning. But the limited number of samples in the new class would cause inaccurate and unstable measurements on visual analogy in high-dimensional representation space, and how to transfer the classification parameters from selected base classes to a new class is also highly non-trivial for the generalization efficacy.

To address the above problems, we first propose a novel Visual Analogy Network Embedded Regression (VANER) model to jointly learn a low-dimensional embedding space and a linear mapping function from the embedding space to classification parameters for base classes. In particular, we learn a low dimensional embedding for each base class with the constraint of embedding similarity between two base classes being able to reflect their visual analogy in the original representation space. Meanwhile, we learn a linear mapping function from the embedding of a base class to its previously learned classification parameters (*i.e.* the logistic regression parameters). The VANER model enables the transformation from original representation space to embedded space and further into classification parameters. We then propose an out-of-sample embedding method to learn the embedding of a new class represented by a few samples through its visual analogy with base classes. By inputting the learned embedding into VANER, we can derive the classification parameters for the new class. Note that these classification parameters are purely generalized from base classes (*i.e.* transferred classification parameters), while the samples in the new class, although only a few, can also be exploited to generate a set of classification parameters (*i.e.* model classification parameters). Therefore, we further investigate the fusion strategy of the two kinds of parameters so that the prior knowledge and data knowledge can be fully leveraged. The framework of the proposed method is illustrated in Figure 1.

We intensively evaluate the proposed method, and the results show that our method can reach 0.87 AUC in average in 200 new classes from ImageNet in one-shot setting (*i.e.* each new class only consists of 1 image sample). In contrast, the AUC of logistic regression with only endogenous parameters is 0.77. We also find that improvement margins (between our method and baselines) in different new classes have significant positive correlation with the relative similarity ratio between a new class and base classes, indicating that our method is consistent with human-like learning.

The technical contributions of this paper are three folds.

- We study the problem of learning to learn from a new angle: given fixed representation space, how to generalize the classification parameters of base classes to a new class? This problem setting can promote new research attempts towards human-like learning mechanism.
- We propose to use visual analogy as the bridge for classification parameter generalization across different classes, and propose a novel VANER method to achieve the transformation from original representation to classification parameters for any new class.
- We intensively evaluate the proposed method and the results show that our method consistently and significantly outperform other baselines, and, more importantly, our method is more consistent with human-like learning.

The rest of the paper are organized as follows. In Section 2, we briefly review the related work of the image classification problem especially on low-shot problem and the network embeddings. In Section 3, we present the framework of our VANER model. In Section 4, we discuss about the experimental results. Finally, we conclude the paper with a discussion of our findings and open issues in section 5.

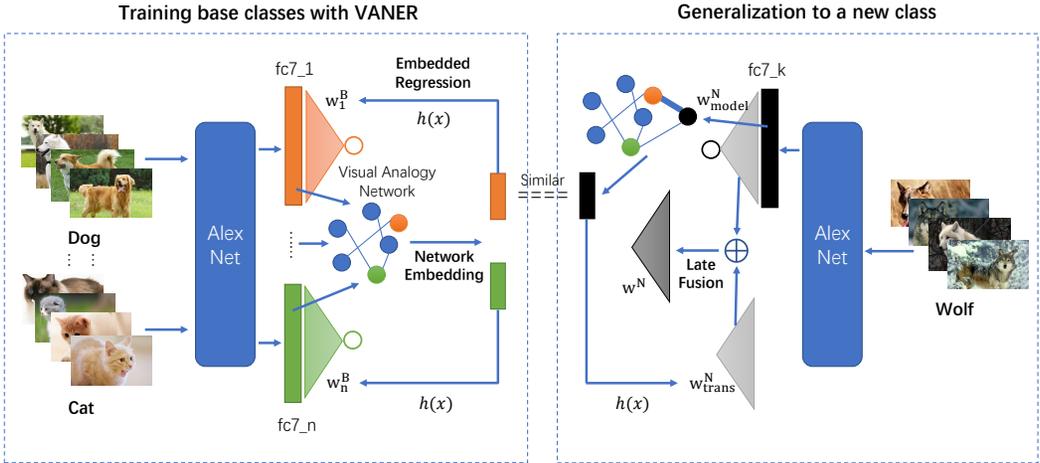


Fig. 1. The framework of learning to learn image classifiers. *Training Base Classes with VANER*: By training base classes with VANER, we derive the embeddings of each base class and the common mapping function from embeddings to classification parameters. *Generalization to a New Class*: Given a new class with only a few samples, we can infer its embedding through out-of-sample inference, and then transform the embedding into transferred classification parameters by the mapping function learned by VANER. After training the classifier with new class samples and getting the model classification parameters, we fuse the two kinds of parameters to form the final classifier.

2 RELATED WORK

The related works can be categorized in three lines, including image classification with deep learning, one/low-shot image classification and network embedding, which we briefly review and discuss as follow.

Image Classification with Deep Learning. The first paper concentrating on the task of image classification using deep convolutional neural network on large-scaled image dataset dates back to the AlexNet [7] in 2012, which reaches an error rate of 17.0% of top-5 prediction on the ILSVRC2010 dataset. After the huge success of the deep neural network on image classification, more and more complex network structures are constantly put forward. Among them, VGGNet[15], GoogLeNet[16] and the ResNet[5] are well known and they reached an error rate of 6.8%, 6.67%, 3.57% on top-5 prediction on the same dataset respectively. All of them are end-to-end models with tons of parameters, leading to its disadvantage of data-hungry.

One/Low-shot Image Classification. One/Low-shot image classification problem mainly focuses on how to learn much information about a category from just one, or a handful of images instead of the large-scaled training dataset. Most of one/low-shot image classification algorithms take advantage of transfer learning. In the early work, [2] proposed a transfer method via a Bayesian approach on the low-level feature of the images. Due to the effectiveness SVM in image classification, many methods are proposed to combine the SVM parameters of the base classes to learn for the transfer parameter of one-shot classes. [11, 22] propose a transfer mechanism using Adaboost method. They both construct a set of weak classifiers through the data from the base classes, and learn a new classifier by linearly combining the weak classifiers. [18] proposes an adaptive Least-Square SVM method to directly combine the base classes SVM model and learn the weights automatically. These methods cannot work well on one-shot problem, as they require

sufficient supervised information to learn the weight of the combined model. Also these methods are based on hand-crafted features, which seriously limits their performance.

After deep learning is introduced into the large-scale image classification, researchers turn to investigate the one-shot problem with deep learning. Some methods are proposed to learn a better image representation to adapt to one-shot image classification problem. [6] introduces a two-way Siamese Neural Network to learn the similarity of two input images using base classes and predict the most possible one-shot class for test images. [4] proposes a Squared Gradient Magnitude Loss considering both the multi-class logistic loss and small dataset training loss. Some other methods combine traditional deep neural network structures with new transfer learning algorithms. [14] uses Memory-Augmented Neural Networks with a Least Recently Used Access module which can be seen as an external memory storing previously learned information, and later [20] proposes an improved method called Matching Network. They both capture the similarity of the novel classes with base classes and utilize the information to do an cross-class transfer, but they optimize the transfer process in representation learning step, rather than classification step. [21] proposes a Model Regression Network for intra-class transfer which learns a nonlinear mapping from the model parameter trained by small-samples to the model parameter trained by large-samples. This mapping can be used to infer the classification parameters via only low-shots (*i.e.* small-samples) in the new classes. But the correlation patterns between small-sample and large-sample parameters are not always notable, which is demonstrated in our experiments. More recently, a few works exploit generative models to create more data for training. [12] takes advantage of the deep generative models to give a method to produce similar images as a given image. [4] then proposes another algorithm to complete the transformation analogy in high-level image features and use this mechanism to expand the images in low-shot classes. Data generation is a feasible way to address the problem of sparse training samples. Differently, our paper attempts to address the problem from the angle of new learning mechanisms.

Network Embedding. In this paper, we exploit network embedding to model visual analogy among different classes, so here we briefly review the recent advances in network embedding. Network Embedding is used to extract the formalized representation of each node in a large-scaled graph. The low-dimension hidden embeddings could capture not only the characteristics of the whole network (*e.g.* the relationship between two nodes) but also the features of the each node itself. Now the network embedding method is widely used in social network area to solve the node clustering or link prediction problems *etc.* There are many algorithms issued to learn the embeddings much better and much faster. [1] uses a matrix factorization technique which is optimized by SGD. [17] proposes LINE method which preserves both the first-order and second-order proximities of each node and improves the quality of the embeddings. Network embedding is proved to be a effective method while dealing with graph analysis.

3 THE METHOD

3.1 Notations and Problem Formulation

Suppose that we have an image set I , and the set is divided into base-class set $I^B = I_1^B \cup I_2^B \cup \dots \cup I_n^B$ which have sufficient training samples, and novel-class set $I^N = I_1^N \cup I_2^N \cup \dots \cup I_m^N$ which have only a few training samples in each class. We train an AlexNet [7] on I^B as our base CNN model and extract its fc7 layer as the high-level features of images. The feature space is denoted as $\mathcal{X} \subset \mathbb{R}^d$. For each image in I^B , we obtain its fc7 layer feature $\mathbf{x}_{ij}^B \in \mathcal{X}$ where $i = 1, 2, \dots, n$ represents its class and $j = 1, 2, \dots, |I_i^B|$ represents its index in class i . We use the same CNN model to derive high-level representations for images in novel classes, denoted by \mathbf{x}_{ij}^N .

A typical binary classifier can be represented as $f(\cdot; \mathbf{w}|\mathbf{X})$ which is a mapping function $f: \mathbb{R}^d \rightarrow \mathbb{R}$ parametrized by \mathbf{w} . The input is a d -dimensional image feature vector and the output is the possibility that the image belongs to the class. We use \mathbf{w}_i^B to denote the parameters for base class i and \mathbf{w}_i^N for novel class i . Based on the above notations, Our problem is defined as follows.

PROBLEM 1 (LEARNING TO LEARN IMAGE CLASSIFIERS). *Given the image features of base classes \mathbf{X}^B , the well-trained base classifier parameters \mathbf{W}^B , and the image features of a novel class i \mathbf{X}_i^N with only a few positive samples, learn the classification parameters \mathbf{w}_i^N for the novel class, so that the learned classifier $f(\cdot; \mathbf{w}_i^N | \mathbf{X}^B, \mathbf{W}^B, \mathbf{X}_i^N)$ can precisely predict labels for the i^{th} novel class.*

Note that the problem of learning to learn image classifiers differs from traditional image classification problems in that the learning of a classifier for a novel class depend on the previously learned base-class classifiers and the image representations in base classes besides the image samples in the novel class.

3.2 Framework of Learning to Learn Image Classifiers

The main idea of our method is to generalize the classification parameters of well-trained base classes to a novel class with only a few training samples. In order to realize this, we propose a framework for learning to learn image classifiers (as shown in Figure 1), which consists of two major steps including (1) learning the mapping function from representation space to classification parameters in base classes and (2) generalizing the base classification parameters to a novel class.

For the first step, we propose a novel VANER model to learn the mapping function. After acquiring the high-level representations from fc7 layer in AlexNet for all images in base classes, we calculate the mean feature vector for each class, and generate a visual analogy network for base classes by measuring their pair-wise class similarity. From the visual analogy network, we learn a low-dimensional embedding for each base class with the constraints that the embeddings of classes should preserve the visual analogy network structures, and, at the same time, the embedding of a base class can be transformed into the classification parameters of the base class through a linear mapping function. By training in base classes, we can derive the embeddings of base classes, and a mapping function from embeddings to classification parameters.

Given a novel class with only one or a few samples, we get its high-level representations through the same AlexNet trained in base classes. By comparing its feature vector with those of base classes, we construct a visual analogy network incorporating the novel classes and base classes, from which we can infer the embedding for the novel class through an out-of-sample embedding method. With the inferred embedding of the novel class and the mapping function learned in VANER, we obtain the classification parameters generalized from base classes. Meanwhile, we also learn the classification parameters for the new class from its samples (although only a few). After that, we conduct late fusions on these two kinds of parameters so that the knowledge from prior knowledge and data are fully leveraged. Finally we use the fused classification parameters to classify the novel class.

The notion of this framework is that the classifier for a novel class should be similar as that for a base class if and only if the novel class is visually analogous with the base class. In the example of Figure 1, the novel class wolf is similar as the base class dog in high-level representations, so the link between them will have a high weight in the visual analogy network. This high-weight link will enforce the embedding of wolf class to be similar with that of dog class, and the similar embeddings will result in similar classification parameters as they share the same mapping function.

In this way, the classification parameters of the dog class which is well trained with sufficient training data can be successfully transferred to the new wolf class.

3.3 The VANER Model

We define a network $G = (V, E)$ where V is the vertex set of the graph, with each vertex representing a base class and $|V| = n$. E is the edge set of the graph, each edge represents visual analogy relationship between two classes with the edge weight depicting the similarity degree. We use \mathbf{A} to represent the adjacency matrix of the network, and A_{ij} is the edge weight from vertex i to vertex j . $\mathbf{A}_{i, \cdot}$ and $\mathbf{A}_{\cdot, j}$ stands for the i -th row and the j -th column of \mathbf{A} respectively. In our classification problem, we construct the visual analogy network as a undirected full-connected graph, and edge weight (i.e. degree of visual analogy) between two classes is calculated by:

$$A_{ij} = \frac{\overline{\mathbf{x}}_i^B \cdot \overline{\mathbf{x}}_j^B}{\|\overline{\mathbf{x}}_i^B\|_2 \cdot \|\overline{\mathbf{x}}_j^B\|_2}. \quad (1)$$

Here $\overline{\mathbf{x}}_i^B$ means the average feature vector for class i and this equation is the cosine distance between two base classes. Note that our graph is an undirected graph, and the adjacency matrix \mathbf{A} is symmetric.

In order to make the visual analogy measurement robust in sparse scenarios, we need to reduce the representation space dimensions. Our basic hypothesis in generalizing classification parameters is that if two class are visual similar, they should share similar classification parameters. We realize this by imposing a linear mapping function from the embedding space to classification parameter space, so that similar embeddings will result in similar classification parameters. Motivated by this, we propose a Visual Analogy Network Embedded Regression model.

Let $\mathbf{V} \in \mathbb{R}^{n \times q}$ be the embeddings for all nodes in the network, and each row of \mathbf{V} with dimension q is the embedding for each vertex. Let $\mathbf{W} \in \mathbb{R}^{n \times p}$ represent all parameters of the base classifiers. There is also a common linear transformation matrix for all base classes $\mathbf{T} \in \mathbb{R}^{q \times p}$ to convert the embedding space to the classification parameter space for all base classifiers. Then the loss function is defined as:

$$\mathcal{L}(\mathbf{V}, \mathbf{T}) = \|\mathbf{V}\mathbf{T} - \mathbf{W}\|_F^2 + \lambda \|\mathbf{A} - \mathbf{V}\mathbf{V}^\top\|_F^2. \quad (2)$$

where $\|\cdot\|_F$ is the Frobenius Norm of the matrix.

The first term enforces the embeddings to be able to converted into the classification parameter through a linear transformation. The second term constrain the embeddings to preserve the structure of the visual analogy network. Our goal is to find the matrix \mathbf{V} and \mathbf{T} to minimize this loss function.

This is a common unconstrained two variables optimization problem and we use the alternative coordinate descent method to find the best solution for \mathbf{V} and \mathbf{T} , where the gradients are calculated by:

$$\begin{cases} \frac{\partial \mathcal{L}(\mathbf{V}, \mathbf{T})}{\partial \mathbf{V}} = 2(\mathbf{V}\mathbf{T} - \mathbf{W})\mathbf{T}^\top + \lambda(-4\mathbf{A}\mathbf{V} + 4\mathbf{V}\mathbf{V}^\top\mathbf{V}) \\ \frac{\partial \mathcal{L}(\mathbf{V}, \mathbf{T})}{\partial \mathbf{T}} = 2\mathbf{V}^\top(\mathbf{V}\mathbf{T} - \mathbf{W}). \end{cases} \quad (3)$$

3.4 Embedding Inference for Novel Classes

By training VANER model in base classes, we can get the embeddings for each base class and the mapping function from embeddings to classification parameters. Given a new class with only a few samples, we need to infer its embedding. Suppose the embedding for the novel class is $\mathbf{v}_{new} \in \mathbb{R}^q$.

We calculate the similarity of a novel class with all base classes by Equation 1, and we denote this similarity vector by $\mathbf{a}_{new} \in \mathbb{R}^n$.

Then we define the objective function for the novel class embedding inference and our goal is to minimize the following function:

$$\mathcal{L}(\mathbf{v}_{new}) = \left\| \begin{bmatrix} \mathbf{A} & \mathbf{a}_{new}^\top \\ \mathbf{a}_{new} & 1 \end{bmatrix} - \begin{bmatrix} \mathbf{V} \\ \mathbf{v}_{new} \end{bmatrix} \begin{bmatrix} \mathbf{V}^\top & \mathbf{v}_{new}^\top \end{bmatrix} \right\|_F^2. \quad (4)$$

Equation 4 is in fact the extension of the second term in Equation 2. As we have little information about the classification parameters of the novel class, we omit the second term in Equation 2.

After we delete the independence term of \mathbf{v}_{new} , the final minimization problem for us to solve is:

$$\min \mathcal{L}(\mathbf{v}_{new}) = 2 \left\| \mathbf{a}_{new} - \mathbf{v}_{new} \mathbf{V}^\top \right\|_2^2 + (\mathbf{v}_{new} \mathbf{v}_{new}^\top - 1). \quad (5)$$

In fact, the second term of Equation 5 is a regular term. We omit the second term and thus the first term is in the form of a linear regression loss. Then we can get the explicit solution for \mathbf{v}_{new} without using gradient descent. The solution is represented as:

$$\mathbf{v}_{new} = \mathbf{a}_{new} (\mathbf{V}^\top)^+, \quad (6)$$

where \mathbf{M}^+ is the Moore-Penrose pseudo-inverse of matrix \mathbf{M} defined by $(\mathbf{M}^\top \mathbf{M})^{-1} \mathbf{M}^\top$. Note that we could speed up the algorithm by pre-computing the pseudo-inverse of \mathbf{V}^\top .

After deriving the embedding for the new class, we can easily obtain its transferred classification parameters by multiplying transformation matrix \mathbf{T} :

$$\mathbf{w}_{new}^N = \mathbf{v}_{new} \mathbf{T}. \quad (7)$$

3.5 Late Fusion

As mentioned above, we can also learn the classification parameters of a new class from its samples (although only a few), and we call them model classification parameters. Then we need to fuse the transferred classification parameters and model classification parameters into the final classifier. Here we present three strategies for late fusion: Initializing, Tuning, and Voting.

Let $f(\cdot, \mathbf{w}^N) : \mathbb{R}^d \rightarrow [0, 1]$ be the binary classifier for a new class. \mathbf{X}_T is the mixture set of positive and negative samples, and y is the label with $y = 1$ indicating positive sample and $y = 0$ indicating negative sample.

Initializing We use the transferred classification parameters as an initialization and then re-learn the parameters of new classifier by the new class samples. The training loss function is defined as the common loss function for classification. That is:

$$\mathcal{L}(\mathbf{w}^N) = \left\{ \sum_{\mathbf{x} \in \mathbf{X}_T} L(f(\mathbf{x}, \mathbf{w}^N), y) \right\} + \lambda \cdot R(\mathbf{w}^N), \quad (8)$$

where $L(\cdot, \cdot)$ is the prediction error and we use cross-entropy loss in our experiment. $R(\cdot)$ is a regularization term and we use $L2$ -norm in our experiment. For learning \mathbf{w}^N , we use the batched Stochastic Gradient Descent (SGD) and the \mathbf{w}^N is initialized with the transferred classification parameters \mathbf{w}_{trans}^N .

Tuning We train the model classification parameters with new class samples, while adding a loss term to constrain the similarity of the transferred classification parameters and the final parameter:

$$\mathcal{L}(\mathbf{w}^N) = \left\{ \sum_{\mathbf{x} \in X_T} L(f(\mathbf{x}, \mathbf{w}^N), y) \right\} + \lambda \cdot \|\mathbf{w}^N - \mathbf{w}_{trans}^N\|_F^2. \quad (9)$$

Here, \mathbf{w}_{trans}^N is the transferred parameter we obtain from the previous steps (*i.e.* \mathbf{w}_{new}^N in Equation 7). We still use the batched SGD method with a random initialization to solve for \mathbf{w}^N .

Voting This method is a weighted average for the transferred classification parameters and the learned model classification parameters. First, we learn a \mathbf{w}_{model}^N using the Equation 8 with random initialization. Then we get the final parameter by:

$$\mathbf{w}^N = \mathbf{w}_{trans}^N + \lambda \cdot \mathbf{w}_{model}^N. \quad (10)$$

The hyper-parameter λ serves as a voting weight.

3.6 Complexity Analysis

During the training process of our VANER model, the main cost is to calculate the gradient of the loss function $\mathcal{L}(\mathbf{V}, \mathbf{T})$. For calculating the first derivative of \mathcal{L} with respect to \mathbf{V} , the complexity per iteration is $O(nq \cdot \max(p, n))$. As to the first derivative of \mathcal{L} with respect to \mathbf{T} , the complexity per iteration is $O(nq \cdot \max(p, q))$. While predicting the novel class, if we use Equation 6 for accelerating, we are able to pre-compute the $(\mathbf{V}^\top)^+$ for $O(nq^2)$ and for each novel class, the complexity of the predicting process is $O(q \cdot \max(p, n))$.

4 EXPERIMENTS

4.1 Data and Experimental Settings

In our experiments, we mainly use the ILSVRC2015 dataset [13], whose training set contains over 1.2 million images in 1,000 categories. We randomly divide the ILSVRC training dataset into 800 base classes and 200 novel classes. We retrain the AlexNet on the 800 base classes as our base CNN model. Before training, each image is cropped into 227×227 size and all of the training setting is the same as [7] except that we do not use the data augmentation method. After training, we use the fc7 layer of AlexNet as the high-level representations for images.

Our goal is to learn the classifier for a new class given the base classifiers. So we set our problem to be a binary classification problem, where the new classifier is learned to classify the novel class (as positive samples) and all the base classes (as negative samples). In training phase, we randomly select k images as the training set for each novel class to simulate k -shot learning scenario. In testing phase, given a novel class, we randomly select 500 images (with no overlap with the training set) from it as the positive examples and randomly select 5 images from each base class of the ILSVRC2015 validation set as negative samples. To eliminate randomness, for any k -shot setting, we run 10 times and report the average result in the following experiments.

The evaluating metric in our experiment is the Area Under Curve (AUC) of the Receiver Operating Characteristic (ROC) and the F1-score, which are widely used in binary classification.

4.2 Baseline

We compare our method with the baselines below. We divide these algorithms into three categories: The first algorithm is the traditional method used for image classification; the next two algorithms are the methods mainly used in one-shot image classification, in accordance with our algorithm's setting, we choose those algorithms which learn a new classifier while keeping the features of

the image unchanged and among them MRN[21] is state-of-the-art; the last three algorithms are within our framework but certain parts of the whole algorithm are excluded for comparison. In order to demonstrate the characteristics and advantages of our method, we also implement these variational versions of our method.

Logistic Regression (LR) We directly use the the novel class images as positive training samples and the randomly selected base class images as negative samples to train a logistic regression classifier. It is regarded as the null model without any generalization from the base classifiers.

Weighted Logistic Regression (Weighted-LR) Here we use the weighted average of the base classifiers' parameters as the classification parameters for the new class. The weights are calculated by a L_2 -normalization of cosine similarities between the feature vector of the novel class and those of all base classes. This method share a similar notion to transfer base classifiers to novel classes, but the transferring process is heuristic.

Model Regression Networks (MRN) [21] This method suppose that there is a mapping function from the classification parameters trained with small samples to those trained with large samples within the same class, and this mapping function can be learned from base classes. Then, given a new classifier trained with small samples, the learned MRN is reused to predict the classification parameters trained with large samples.

VANER We only use the classification parameters transferred from base classes to classify the new classes, and do not consider the parameters generate by new class samples. This method is designed to demonstrate the importance of late fusion.

VANER(-Mapping) We directly learn the embedding by Equation 2 without the first regression term. Then we use the above weighted-LR method in the embedding space instead of the original feature space. This method is used to evaluate the effectiveness of the mapping function.

VANER(-Embedding) We directly train a regression model from the original feature space to the classification parameter space without the network embedding. This method is used to demonstrate the effectiveness of class node embedding on the visual analogy network.

4.3 Results

4.3.1 Classification Performance on Novel Classes. In this section, we evaluate how well the classifiers learned by our method and other baselines can perform in new classes. The results are shown in Table 1. We can see that in all the low-shot settings, our method *VANER + Voting* consistently performs the best in both *AUC* and *F1* metrics. In contrast, *LR* performs the worse in 1-shot setting, which demonstrate the importance of generalization from base classes when the new class has very few samples. *MRN* does not work well in most settings, demonstrating that its basic hypothesis that the classification parameters trained by large samples and small samples respectively are correlated do not necessarily hold in real data. By comparing *VANER + Voting* with the other three variational versions of our method, we can safely draw the conclusion that the major ingredients in our method, including network embedding for low dimensional representations, mapping function for transforming embedding space to classification parameter space, as well as the late fusion strategy are necessary and effective. We also compare different late fusion strategies. From the results shown in Table 2 we find that the Voting strategy is more fit for our scenario.

Furthermore, we compare the performances of these methods in different low-shot settings, and the results are shown in Figure 2. We can see that our method consistently performs the best in all settings, and the advantage of our method is more obvious when the new classes have less training samples. In particular, by comparing our method and *LR*, we can see that *LR* need about 20 shots to reach *AUC* 0.9, while we only need 2 shots, indicating that we can save 90% training data. An

Table 1. Performance of different algorithms for k -shot problem

Algorithm	1-shot		5-shot		10-shot		20-shot	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1
<i>VANER + Voting</i>	0.8718	0.5671	0.9425	0.7039	0.9543	0.7343	0.9607	0.7510
<i>VANER</i>	0.8556	0.5292	0.9271	0.6491	0.9379	0.6721	0.9432	0.6850
<i>VANER(-Mapping)</i>	0.8261	0.4551	0.8526	0.4807	0.8726	0.5179	0.8897	0.5394
<i>VANER(-Embedding)</i>	0.7922	0.4335	0.9032	0.6015	0.9183	0.6347	0.9393	0.6788
<i>LR</i>	0.7705	0.3994	0.8885	0.5882	0.9134	0.6421	0.9341	0.6877
<i>Weighted - LR</i>	0.8338	0.4680	0.8350	0.4691	0.8374	0.4711	0.8411	0.4726
<i>MRN</i>	0.8083	0.4511	0.9175	0.6653	0.9361	0.7133	0.9474	0.7388

Table 2. Performance of different late fusion mechanism for k -shot problem

Algorithm	1-shot		5-shot		10-shot		20-shot	
	AUC	F1	AUC	F1	AUC	F1	AUC	F1
<i>VANER</i>	0.8556	0.5292	0.9271	0.6491	0.9379	0.6721	0.9432	0.6850
<i>VANER + Initializing</i>	0.7662	0.3941	0.9030	0.6185	0.9338	0.6887	0.9461	0.7237
<i>VANER + Tuning</i>	0.7923	0.4244	0.9098	0.6307	0.9365	0.7012	0.9466	0.7268
<i>VANER + Voting</i>	0.8718	0.5671	0.9425	0.7039	0.9543	0.7343	0.9607	0.7510

interesting phenomenon is that the performance of *Weighted - LR* do not change with the shot number increasing. The main reason is that the heuristic rule is not flexible enough to incorporate new information. This demonstrate the importance of learning to learn, rather than rule-based learning.

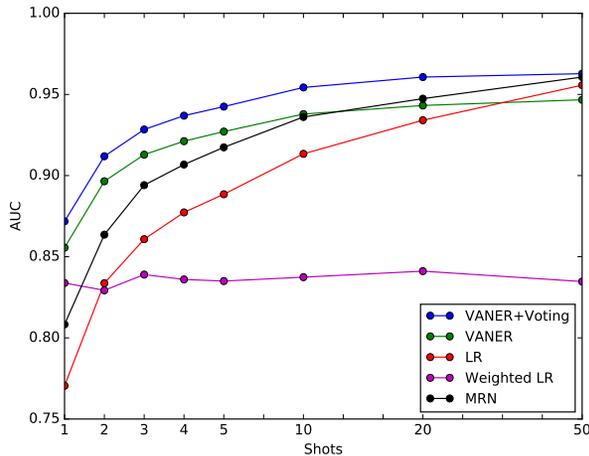


Fig. 2. The change of performance as the shots number increases.

Table 3. Comparison of the our method and *LR* over in novel classes with 1-shot setting

Category	LR (No Transfer)	VANER (Transfer)
Jeep	0.8034	0.9469
Zebra	0.8472	0.9393
Hen	0.7763	0.8398
Lemon	0.6854	0.9583
Bubble	0.7455	0.7041
Pineapple	0.7364	0.8623
Lion	0.8305	0.9372
Screen	0.7801	0.9056
Drum	0.6510	0.6995
Restaurant	0.7806	0.8787

4.3.2 Insightful Analysis. Although our method performs the best in various settings, the failure cases are easy to find. We are interested in the following questions: (1) what are the typical failure cases? (2) what is the driving factor that controls the success of generalization? and (3) is the generalization process explainable?

In order to answer the above questions, we further conduct insightful analysis. Firstly, we randomly select 10 novel classes, and list the performance of our method and *LR* in one-shot setting on these classes, as shown in Table 3. We can see that the effect of generalization is very obvious in 9 classes, while in the bubble class, the generalization plays a negative role.

To discover the driving factor controlling success and failure, we define and calculate the similarity ratio (SR) of a novel class with the base classes by:

$$SR = \frac{\text{Average Top-}k \text{ Similarity with Base Classes}}{\text{Average Similarity with Base Classes}} \quad (11)$$

Here the similarity of two classes is calculated by Equation 1. Intuitively, if a new class is very similar with the top-*k* base classes, while dissimilar with the remained base classes, its Similarity Ratio will be high.

In this experiment, we do a linear regression of the relative improvement in *AUC* of our method over the non-transfer method *LR* in 1-shot setting on the Similarity Ratio for each novel class. The dependent variable indicates the success degree of generalization. And we use $k = 10$ as our experiment setting. We plot the similarity ratio and relative improvement of all new classes in Figure 3. We can see the relative improvement in a new class is positively correlated with the similarity ratio of the new class, with 95% confidence interval for the correlation coefficient range between 0.124 and 0.169.

The results fully demonstrate that our method is consistent with human-like learning: First, the faster we can learn a new concept if it is more similar with some previously learned concepts. (*i.e.* Leading to the increase of the numerator of the Similarity Ratio). Second, the faster we can learn a new concept if we have learned more diversified concepts (*i.e.* Leading to the decrease of the denominator of the Similarity Ratio). This principle can also be used to guide the generalization process and help to determine whether a new class is fit for generalization.

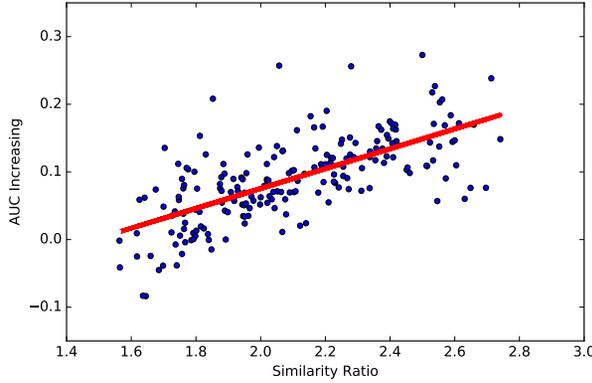


Fig. 3. AUC improvement v.s. Similarity ratio for all novel classes

Novel Class	 Jeep	 Zebra	 Lemon	 Lion	 Screen	 Restaurant
Top-3 Similar Base Classes	 Pickup  Beach_wagon  Tow_truck	 Echidna  Leopard  Cheetah	 Orange  Acorn  Granny_Smith	 Cougar  Dingo  Lynx	 Monitor  Laptop  Television	 Shoe_shop  Marimba  Bakery

Fig. 4. Top-3 most similar base classes to novel class on embedding layer in 5-shot setting.

Finally, we validate whether the generalization process is explainable. Here we randomly select 6 novel classes, and for each novel class, we visualize the top-3 base classes that are most similar with the novel class, as shown in Figure 4. In our method these base classes have large impact on the formation of the new classifier. We can see that the top-3 base classes are visually correlated with the novel classes, and the generalization process can be very intuitive and explainable.

4.3.3 *Parameter Analysis.* In our method, there are two important parameters: voting parameter and the number of embedding dimension. The voting parameter decides the relative weights of the transfer parameters and model parameters in the fusion stage. Here we fix an 1-shot/5-shot/20-shot setting and observe the change of the performance as we tune the voting parameter. The result is shown in Figure 5. We can see that the voting parameter is relatively stable consisting in different settings, so we use 0.2 as the parameter for all k -shot settings. We also tune the number of embedding dimensions and observe the performance change. The results are shown in Figure 6. We can see that there is a large stable range that we can select, and we select 600 in our experiments.

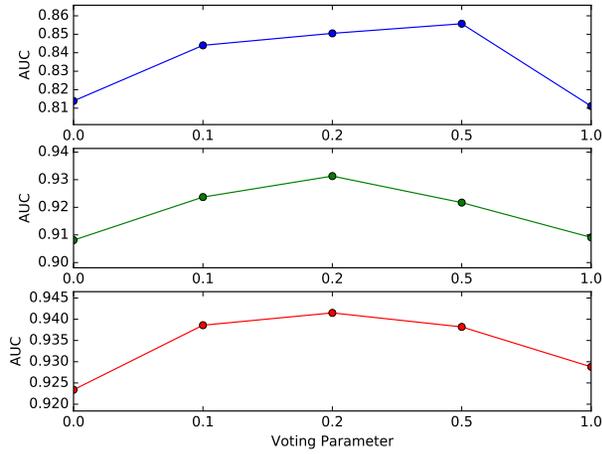


Fig. 5. Parameter analysis on the voting parameter. Top: 1-shot setting, Middle: 5-shot setting, Bottom: 10-shot setting.

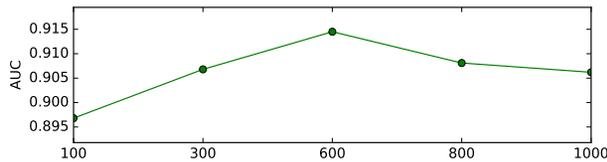


Fig. 6. Parameter analysis on the number of embedding dimensions.

5 CONCLUSIONS AND FUTURE WORKS

In this paper, we investigate the problem of learning to learn image classifiers and attempt to explore a new human-like learning mechanism which fully leveraged the previously learned concepts to assist new concept learning. In particular, We organically combine the ideas of learning to learn and learning by analogy and propose a novel VANER model to fulfill the generalization process from base classes to novel classes. From the extensive experiments, we can safely draw the conclusion that the proposed method performs much better than baselines, complies with human-like learning and provide insightful and intuitive generalization process.

REFERENCES

- [1] Amr Ahmed, Nino Shervashidze, Shравan Narayanamurthy, Vanja Josifovski, and Alexander J Smola. 2013. Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web*. ACM, 37–48.
- [2] Li Fei-Fei, Rob Fergus, and Pietro Perona. 2006. One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence* 28, 4 (2006), 594–611.
- [3] Dedre Gentner and Keith J Holyoak. 1997. Reasoning and learning by analogy: Introduction. *American Psychologist* 52, 1 (1997), 32.
- [4] Bharath Hariharan and Ross Girshick. Low-shot Visual Recognition by Shrinking and Hallucinating Features. (????).
- [5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.
- [6] Gregory Koch. 2015. *Siamese neural networks for one-shot image recognition*. Ph.D. Dissertation. University of Toronto.

- [7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.
- [8] Roland Kwitt, Sebastian Hegenbart, and Marc Niethammer. 2016. One-shot learning of scene locations via feature trajectory transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 78–86.
- [9] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. 2016. Building machines that learn and think like people. *arXiv preprint arXiv:1604.00289* (2016).
- [10] Novi Patricia and Barbara Caputo. 2014. Learning to Learn, from Transfer Learning to Domain Adaptation: A Unifying Perspective. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [11] Guo-Jun Qi, Charu Aggarwal, Yong Rui, Qi Tian, Shiyu Chang, and Thomas Huang. 2011. Towards cross-category knowledge propagation for learning visual concepts. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE, 897–904.
- [12] Danilo Jimenez Rezende, Shakir Mohamed, Ivo Danihelka, Karol Gregor, and Daan Wierstra. 2016. One-shot generalization in deep generative models. *arXiv preprint arXiv:1603.05106* (2016).
- [13] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)* 115, 3 (2015), 211–252. DOI: <http://dx.doi.org/10.1007/s11263-015-0816-y>
- [14] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. 2016. One-shot learning with memory-augmented neural networks. *arXiv preprint arXiv:1605.06065* (2016).
- [15] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).
- [16] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1–9.
- [17] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. 2015. Line: Large-scale information network embedding. In *Proceedings of the 24th International Conference on World Wide Web*. ACM, 1067–1077.
- [18] Tatiana Tommasi, Francesco Orabona, and Barbara Caputo. 2014. Learning categories from few examples with multi model knowledge transfer. *IEEE transactions on pattern analysis and machine intelligence* 36, 5 (2014), 928–941.
- [19] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. 2015. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*. 4068–4076.
- [20] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Daan Wierstra, and others. 2016. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*. 3630–3638.
- [21] Yu-Xiong Wang and Martial Hebert. 2016. Learning to learn: Model regression networks for easy small sample learning. In *European Conference on Computer Vision*. Springer, 616–634.
- [22] Yi Yao and Gianfranco Doretto. 2010. Boosting for transfer learning with multiple sources. In *Computer vision and pattern recognition (CVPR), 2010 IEEE conference on*. IEEE, 1855–1862.
- [23] Xiangxin Zhu, Dragomir Anguelov, and Deva Ramanan. 2014. Capturing Long-tail Distributions of Object Subcategories. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Received September 2017