DeepME: Deep Mixture Experts for Large-scale Image Classification

Ming He[1,2,4], **Guangyi Lv**[2†], **Weidong He**[2], **Jianping Fan**[3], **Guihua Zeng** [1*]
[1]Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[2]University of Science and Technology of China, Hefei, Anhui 230000, China
[3]AI Lab at Lenovo Research, Beijing, China
[4]Didi Chuxing, Beijing, China
heming01@foxmail.com, {gylv, hwd}@mail.ustc.edu.cn, jfan1@lenovo.com, ghzeng@sjtu.edu.cn

Abstract

Although deep learning has demonstrated its outstanding performance on image classification, most well-known deep networks make efforts to optimize both their structures and their node weights for recognizing fewer (e.g., no more than 1000) object classes. Therefore, it is attractive to extend or mixture such well-known deep networks to support large-scale image classification. According to our best knowledge, how to adaptively and effectively fuse multiple CNNs for large-scale image classification is still under-explored. On this basis, a deep mixture algorithm is developed to support large-scale image classification in this paper. First, a soft spectral clustering method is developed to construct a two-layer ontology (group layer and category layer) by assigning large numbers of image categories into a set of groups according to their inter-category semantic correlations, where the semantically-related image categories under the neighbouring group nodes may share similar learning complexities. Then, such two-layer ontology is further used to generate the task groups, in which each task group contains partial image categories with similar learning complexities and one particular base deep network is learned. Finally, a gate network is learned to combine all base deep networks with fewer diverse outputs to generate a mixture network with larger outputs. Our experimental results on ImageNet10K have demonstrated that our proposed deep mixture algorithm can achieve very competitive results (top 1 accuracy: 32.13%) on large-scale image classification tasks.

1 Introduction

As we know, by learning high-level features and a N-way softmax in an end-to-end multi-layer manner, deep learning [LeCun *et al.*, 1998; Sun *et al.*, 2014; Sun *et al.*, 2019; Zhang *et al.*, 2019a; Ma *et al.*, 2020] has adequately demonstrated its outstanding performance on classification because of its strong ability on learning highly invariant and discriminative features. Unfortunately, most well-known deep networks [Krizhevsky *et al.*, 2012; Simonyan and Zisserman, 2014; Howard *et al.*, 2019] optimize both their structures (i.e., numbers of layers and units in each layer) and their node weights for recognizing fewer (e.g., $\leq$ 1,000) object classes. Thus it is very attractive to extend such deep networks (for 1,000 object classes) to support large-scale image classification.

Since both the high-level features for image content representation and the $1,000$-way softmax for image classification are trained jointly in an end-to-end fashion, simply enlarging the final outputs of some well-designed deep CNNs (from $1,000$-way softmax into N-way one, $N \geq 10,000$) may not be an optimal solution for large-scale image classification. One potential solution is to use deep mixture [Jacobs *et al.*, 1991; Tang *et al.*, 2012; Masoudnia and Ebrahimpour, 2014; Ge *et al.*, 2016] to combine a set of base deep CNNs. Unfortunately, all the existing deep mixture techniques focus on combining multiple deep CNNs which are trained to classify images into the same set of object classes, e.g., they just combine multiple base deep CNNs with the same task space (same list of outputs). Without supporting effective combination of multiple base deep CNNs with diverse outputs, we cannot leverage the well-designed deep CNNs for 1,000 object classes to support large-scale image classification, e.g., classifying images into tens of thousands of categories. Obviously, it is not straightforward to combine a set of base deep CNNs which are trained to classify images into different subsets of tens of thousands of categories (i.e., their task spaces are different and their outputs are diverse).

According to the best of our knowledge, there is only one work [Zhao *et al.*, 2018] focusing on combining a set of base deep CNNs, which are originally trained to classify images into different subsets of tens of thousands of atomic object classes. A deep mixture of diverse experts algorithm (DMDE, for short) is developed for seamlessly combining a set of base deep networks (i.e., AlexNet) with diverse 1,001-D outputs to generate a mixture network with 7,756-D outputs under the help of a proposed stacking function. With this method, DMDE has been proved to achieve satisfactory performance, however, the design of the stacking function is based on several intuitive observations, which are meaningful but may be a little bit ad-hoc.

Inspired by these observations, in this paper, a new deep

[†]Ming He and Guangyi Lv contributed equally to this work.
[*]Guihua Zeng is the corresponding author.

mixture algorithm is developed to support large-scale image classification. To be specific, it contains the following key components: (a) a soft spectral clustering method is developed to construct a two-layer ontology (group and category layer) by assigning large numbers of image categories into a set of groups (with certain degrees of inter-group overlapping) according to their inter-category semantic correlations. It is worth emphasizing that the overlapping among tasks are adaptively generated, which could well support inter-group message passaging among different tasks; (b) such two-layer ontology is used to generate a set of task groups, where each task group contains fewer image categories with similar learning complexities and one particular base deep network is learned; (c) a gate network is learned to combine all these base deep networks with fewer diverse outputs to generate a mixture network with much larger outputs.

2 Related Research

In this section, we review the most relevant researches on deep learning and mixture of deep CNNs respectively.

2.1 Deep Learning

By learning high-level features and a N-way softmax jointly in an end-to-end multi-layer manner, deep learning [Hinton *et al.*, 2006; Jarrett *et al.*, 2009; Krizhevsky *et al.*, 2012; Sun *et al.*, 2014; Borisyuk *et al.*, 2018; Zhu *et al.*, 2020; Wang *et al.*, 2020; Li *et al.*, 2020] has demonstrated its outstanding performance on significantly boosting the accuracy rates for many tasks. Most well-known deep CNNs (such as AlexNet, VGG and ResNet with 1,000-D outputs) optimize both their network structures (numbers of layers and units in each layer) and their node weights for classifying images into 1,000 object classes, and they cannot simply be used to support large-scale image classification (i.e., which has to recognize tens of thousands of image categories). One intuitive solution is to simply enlarge the network structures (such as enlarging the widths and depths of the CNNs) to configure huge deep CNNs with tens of thousands of outputs, but it may require huge computation cost because the choices of the optimal network structures have historically been relegated to manual optimization, which relies in human intuition and domain knowledge in conjunction with extensive trials and errors. It is worth noting that when more layers and more units per layer are used to configure a huge deep CNNs, the number of node weights being fitted will increase dramatically, thus the number of training samples for each image category should be increased. In addition, the pursuit for very deep networks (more layers) is met with a diminishing return and increased training difficulty, and widening a network would result in a quadratic growth in both computational cost and memory demand. Thus there is no guarantee that learning huge deep CNNs (with more layers and more units on each layer) can allow us to achieve higher accuracy rates on large-scale image classification. Obviously, simply increasing the softmax outputs (from $1,000$-way softmax into N-way one, $N \geq 10,000$) may not be able to achieve good results because the underlying deep CNNs (learned for 1,000 classes) could be insufficient and inefficient to extract discriminative representations for tens of thousands of image categories.

2.2 Mixture of Deep CNNs

To improve the accuracy rates for image classification, traditional deep mixture techniques aim to combine the predictions from multiple base deep CNNs when they are trained to classify images into the same set of object classes [Jacobs *et al.*, 1991; Tang *et al.*, 2012; Masoudnia and Ebrahimpour, 2014; Ge *et al.*, 2016; Zhao *et al.*, 2018; Rao *et al.*, 2018; Zhang *et al.*, 2019b; Nguyen *et al.*, 2019; Ma *et al.*, 2018; Zhang *et al.*, 2020]. In order to enhance the diversity of the base deep CNNs being combined, they are usually trained over different sample subsets, so that they may make their errors in different ways or compensate each other. Ge et al. [Ge *et al.*, 2016] developed a mixture of deep CNNs (MixDCNN) by partitioning the training images into multiple subsets and learning one particular base deep CNNs for each image subset. In such MixDCNN approach, each of these base deep CNNs concentrated on learning the subtle differences for the same set of object classes, thus all these base deep CNNs shared the same task space. Recently, Zhao et al [Zhao *et al.*, 2018] proposed a deep mixture of diverse experts algorithm by seamlessly combining a set of base deep CNNs with diverse 1,001-D outputs to generate a mixture network with 7,756-D outputs. Unfortunately, the underlying stacking function in DMDE was designed according to some intuitive observations, which were meaningful but may be a little adhoc. On the other hand, our deep mixture algorithm learns a gate network for adaptively combining a set of base deep networks with fewer diverse outputs to generate a mixture network with much larger outputs (i.e., 10,184-D outputs for handling ImageNet10K set with 10,184 image categories).

For the sake of emphasis, we would highlight the main three improvements of our proposed model DeepME compared with DMDE. First of all, in DMDE, each image category is assigned to a particular task group, and the overlapping percent among task groups is set manually, which is lack of adaptive for generating task groups. However, in our paper, the proposed soft spectral clustering method could assign each category to several task groups according to its semantic meaning. What's more, the overlapping part among tasks are adaptively generated, which could better support inter-group message passaging among different task groups. Second, in DMDE, the base deep network only introduces the inter-category visual similarity to characterize the inter-category visual similarities. Nevertheless, DeepME introduces both inter-category visual similarity and semantic similarity, which could characterize the inter-category visual similarity and semantic similarity at the same time. Finally, the design of the stacking function is just based on several intuitive observations in DMDE. In DeepME, the well-designed gate network enables DeepME with three strengths: 1) The gate network could depress the irrelevant base deep networks and promote the relevant base deep networks; 2) The gate network could address the overconfidence issue and guarantee that the best-matched image category for each image has higher probability than others in a reasonable margin; 3) The gate network could guarantee that each base deep network is specialized to recognize different subsets of ImageNet10K. The above three strengths contribute to improving DeepME's classifying performance and robustness significantly.

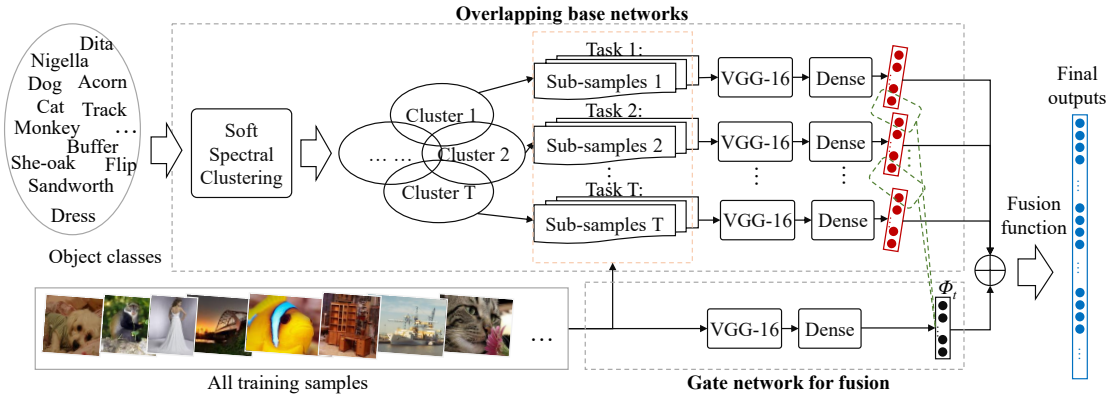

Figure 1: The flowchart of our proposed model, i.e., DeepME.

3 Deep Mixture

To support large-scale image classification, a deep mixture model is developed and it contains the following key components as shown in Figure 1: (a) a Soft Spectral Clustering (SSC) method is first developed to construct a two layer ontology by assigning large numbers of image categories into a set of groups according to their inter-category semantic correlations; (b) such two-layer ontology is used to generate a set of task groups, where each task group contains semantically-related image categories with similar learning complexities; (c) one particular base deep network is learned for each task group and a gate network is learned for combining all base deep networks with fewer diverse outputs to generate a mixture network with much larger outputs.

3.1 Adaptive Task Assignment

We use ImageNet10K data set [Deng *et al.*, 2009] with 10,184 image categories to evaluate our deep mixture algorithm on large-scale image classification. To apply traditional well-known deep networks with 1,000-D outputs over ImageNet10K image set, we first need to partition its 10,184 image categories into a set of task groups. To learn more discriminative base network for each task group and enhance the separability of the image categories in the same task group, we do expect that the image categories with similar learning complexities can be assigned into the same task group, thus it is very attractive to develop new approaches for assigning the semantically-related image categories with similar learning complexities into the same task group. On this basis, a soft spectral clustering method is first developed to construct a two-layer (group layer and class layer) ontology by assigning 10,184 image categories into a set of groups according to their inter-category semantic similarities, where the semantically-related image categories on the neighboring group nodes of our two-layer ontology may share similar learning complexities. Inspired by spectral clustering [Bach and Jordan, 2004] and fuzzy C-means [Bezdek *et al.*, 1984], we develop SSC method, which can assign some uncertain image categories into multiple groups to enable inter-group information sharing and transmission in our deep mixture algorithm.

To support semantic clustering of image categories, we first acquire the similarity matrix $\mathbf{\Theta}$ for all the image categories. As introduced in [Deng *et al.*, 2009], 10,184 image categories in ImageNet10K are organized by the semantic hierarchy of WordNet [Miller, 1995]. Thus a semantic similarity matrix $\mathbf{\Theta}$ is calculated and its element $\theta_{i,j}$ is calculated by the Leacock-Chodorow similarity [Leacock and Chodorow, 1998]:$\theta_{i,j} = -\log \frac{d_{i,j}}{D}$, where $d_{i,j}$ is the shortest path length that connects the category i and the category j over the WordNet ontology, D is the maximum depth of the WordNet ontology in which the category i and j occur.

When the matrix $\mathbf{\Theta}$ is available, the devised SSC method is utilized to partition 10,184 image categories into T clusters (groups): if $p_{c,t} \geq \delta$, the category c is assigned to cluster (group) t, otherwise, the category is not assigned to the cluster (group) t, where $p_{c,t}$ is the soft probability of the category c respect to the cluster (group) t. It is worth noting that both T and δ are determined experimentally via cross-validation. Finally, we acquire 15 groups for ImageNet10K.

To exhibit the performance of our proposed SSC method intuitively, we visualize the clustering results of ImageNet10K in Figure 2, where the blue lines show that these image categories are overlapped by no less than two clusters (groups). From Figure 2, we can draw two main observations: (a) Majority of sub-categories belonging to the same parent category on WordNet have been clustered in the same

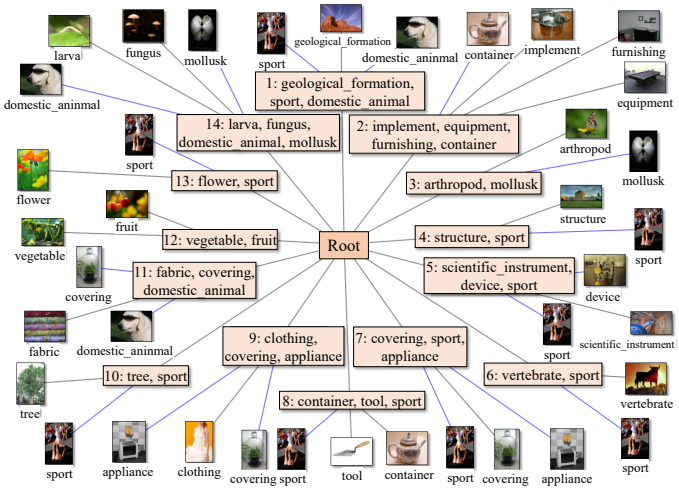

Figure 2: The visualization of ImageNet10K' results.

group, which means that the semantically-related image categories are assigned into the same group; (b) The overlapping categories, which are assigned into multiple groups simultaneously, can establish inter-group correlations and support inter-group message passaging among different task groups.

3.2 Learning Base Deep Networks

Given a task group with C image categories, a base deep network is learned. It is worth noting that all these well-known deep networks can be used to configure our base deep network. In this paper, we directly utilize the network structure of VGG16 [Simonyan and Zisserman, 2014] to configure the base deep network with C outputs. Then, the objective function can be denoted as:

$$\begin{aligned}\min_{\boldsymbol{W}} \quad & \mu\sum_{c=1}^{C}\sum_{l=1}^{R_c}\ell_c^l + \lambda_1 tr(\boldsymbol{W}\boldsymbol{W}^T) \\ & + \frac{\lambda_2}{2}tr(\boldsymbol{W}L(\boldsymbol{\Omega})\boldsymbol{W}^T) + \frac{\boldsymbol{\lambda_3}}{2}tr(\boldsymbol{W}L(\boldsymbol{\Theta})\boldsymbol{W}^T), \quad (1)\\ & \ell_c^l = -\mathbb{I}\{y_c^l\}log\frac{exp(\boldsymbol{w}_c^T VGG(\boldsymbol{x}_c^l)+b)}{\sum_{i=1}^{C}exp(\boldsymbol{w}_i^T VGG(\boldsymbol{x}_i^l)+b)}, \quad (2)\end{aligned}$$

where R_c is the number of images for the category c, $tr(\cdot)$ is the matrix's trace, $\boldsymbol{W} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_C)$ indicates the model's parameters for C categories, μ is the penalty parameter, λ_1, λ_2 and λ_3 represent the regularization terms. ℓ_c^l is the training error rate formulated by the softmax regression. $\mathbb{I}\{y_c^l\}$ is the indication function. If $(\boldsymbol{x}_c^l, y_c^l)$ is the positive training sample for the category c (i.e., $y_c^l = 1$), $\mathbb{I}\{y_c^l\}$ is equal to 1. Conversely, if $(\boldsymbol{x}_c^l, y_c^l)$ is not the positive training sample (i.e., $y_c^l = 0$), $\mathbb{I}\{y_c^l\}$ is equal to 0.

Inspired by [Zhao *et al.*, 2018], we also introduce the $C\times C$ relevant inter-category visual similarity matrix $\boldsymbol{\Omega}$, which is used to characterize the inter-category visual similarities for C categories in the same task group. Except for introducing $\boldsymbol{\Omega}$ to characterize the inter-category visual similarities, we also employ a $C \times C$ relevant inter-category semantic similarity matrix $\boldsymbol{\Theta}$ as prior knowledge to enforce the learning of $\boldsymbol{W}$. The element $\theta_{i,j}$ of $\boldsymbol{\Theta}$ is extracted from $\boldsymbol{\Theta}$ which has been calculated in Section 3.1. Different from the similarity matrix $\boldsymbol{\Omega}$, $\boldsymbol{\Theta}$ is not iteratively updated during the network training process. And L is the Laplacian matrix of the matrix $\boldsymbol{\Omega}$ or $\boldsymbol{\Theta}$. If two image categories i and j have larger inter-category visual or semantic similarity, their model parameters $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ may share some common parts.

Finally, we fine-tune the model by optimizing the above function, i.e., Eq. (1). Then the gradients are back-propagated to refine the weights for the base network.

3.3 Gate Network Training

To generate the network $\mathbb{F}(x)$ with 10,184-D outputs and enhance its discrimination ability, we train: (a) τ base deep networks $\{\boldsymbol{f}_1(x), \cdots, \boldsymbol{f}_t(x), \cdots, \boldsymbol{f}_\tau(x)\}$ with the set of model parameters $\{\boldsymbol{W}_{e,t}\}_{t=1}^{\tau}$; and (b) a τ-D gate network $\boldsymbol{\Theta} = \{\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_t, \cdots, \boldsymbol{\phi}_\tau\}$. Such mixture network $\mathbb{F}(x)$ is defined as:

$$\mathbb{F}(x) = \sum_{t=1}^{\tau}\boldsymbol{\phi}_t\boldsymbol{f}_t(x), \qquad \sum_{t=1}^{\tau}\boldsymbol{\phi}_t^2 = 1, \quad (3)$$

where $\boldsymbol{\phi}_t$ is the confidence score for the t-th base network $\boldsymbol{f}_t(x)$. In consideration of the training/gradient's stability, we apply the L2-norm to ϕ_t. $\boldsymbol{f}_t(x) = \Big\{f_t^1(x), \cdots, f_t^j(x), \cdots, f_t^M(x)\Big\}$ denotes the t-th base network with M-D outputs and $f_t^j(x)$ is the underlying predictor for the j-th image category in the t-th task group G_t. Thus the τ-D gate network $\boldsymbol{\Theta} = \{\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_t, \cdots, \boldsymbol{\phi}_\tau\}$ is learned to determine the individual confidences and contributions of τ base network on generating the mixture network $\mathbb{F}(x)$, and all the images for $N = 10,184$ categories are used to learn the gate network.

Given the training set $\mathcal{D}$ with $R \times N$ i.i.d images that belong to $N = 10,184$ categories in ImageNet10K, $\mathcal{D} = \Big\{\boldsymbol{x}_j^l, y_j^l\Big\}$, $l \in \{1, \cdots, R\}$, $j \in \{1, \cdots, N\}$, the objective function for learning the mixture network is defined as:

$$\begin{aligned}\min_{\boldsymbol{\phi}, \boldsymbol{W}_e} \quad & \mathcal{L}(\mathcal{D}) = \sum_{t=1}^{\tau}\xi(\boldsymbol{W}_{e,t}, \boldsymbol{\phi}_t) + \sum_{t=1}^{\tau}\sum_{h=1}^{\tau}\ell(\boldsymbol{\phi}_t, \boldsymbol{\phi}_h) \\ & + \sum_{l=1}^{R}\sum_{j=1}^{N}\alpha\max\Big(P_{opt}(\boldsymbol{x}_j^l, c_j) - P_{opt}(\boldsymbol{x}_j^l, y_j^l) + \beta, 0\Big), \quad (4)\\ & \xi(\boldsymbol{W}_{e,t}, \phi_t) = \frac{\boldsymbol{\phi}_t F(\boldsymbol{W}_{e,t})}{\sum_{t=1}^{\tau}\boldsymbol{\phi}_t^T\boldsymbol{\phi}_t},\end{aligned}$$

where $\boldsymbol{W}_{e,t}$ is parameters of t-th base deep network, $F(\boldsymbol{W}_{e,t})$ is the loss function of t-th base deep network as shown in Eq.(1), $\boldsymbol{\phi}_t$ and $\boldsymbol{\phi}_h$ are used to indicate the confidence scores for the t-th and h-th base deep network $\boldsymbol{f}_t(x)$ and $\boldsymbol{f}_h(x)$, $\ell(\cdot)$ is the loss function to emphasize the confidence consistency among the predictions from two base deep networks $\boldsymbol{f}_t(x)$ and $\boldsymbol{f}_h(x)$ when they share some common image categories because of inter-group task overlapping, $P_{opt}(\boldsymbol{x}_j^l, c_j)$ is the prediction probability for the training image $\boldsymbol{x}_j^l$ to be identified as the image category c_j and it is aggregated over τ base deep networks, β is a hyper-parameter to denote the confidence margin, α is a hyper-parameter that is used to make trade-off for the importance of the margin-based loss:

$$\begin{aligned}P_{opt}(\boldsymbol{x}_j^l, c_j) &= \sum_{t=1}^{\tau}\mathbb{I}\{y_j^l, c_j\}\boldsymbol{\phi}_t f_t^j(\boldsymbol{x}_j^l), \\ P_{opt}(\boldsymbol{x}_j^l, y_j^l) &= \sum_{t=1}^{\tau}\mathbb{I}\{y_j^l, 1-c_j\}\boldsymbol{\phi}_t f_t^j(\boldsymbol{x}_j^l),\end{aligned} \quad (5)$$

$$\begin{aligned}\boldsymbol{\phi}_t^j &= \frac{1}{R}\sum_{l=1}^{R}\mathbb{I}\{y_j^l, c_j\}\frac{exp(\boldsymbol{W}_{g,t}^T\boldsymbol{x}_j^l + b)}{\sum_{i=1}^{M}exp(\boldsymbol{W}_{g,t}^T\boldsymbol{x}_i^l + b)}, \\ \boldsymbol{\phi}_h^j &= \frac{1}{R}\sum_{l=1}^{R}\mathbb{I}\{y_j^l, c_j\}\frac{exp(\boldsymbol{W}_{g,h}^T\boldsymbol{x}_j^l + b)}{\sum_{k=1}^{M}exp(\boldsymbol{W}_{g,k}^T\boldsymbol{x}_k^l + b)},\end{aligned} \quad (6)$$

$$\ell(\boldsymbol{\phi}_t, \boldsymbol{\phi}_h) = \sum_{c_j \in G_t \cap G_h} H(\phi_t^j, \phi_h^j), \quad (7)$$

where ϕ_t^j is the confidence score for identifying the j-th category in the t-th task group G_t, $\boldsymbol{W}_g$ is the parameters of the gate network, $G_t \cap G_h$ is used to indicate the common set

of the image categories that are shared in two task groups G_t and G_h, $H(\phi_t^j, \phi_h^j)$ is the Hamming distance to measure the dissimilarity between the confidence scores ϕ_t^j and ϕ_h^j for two base deep networks $\boldsymbol{f}_t(x)$ and $\boldsymbol{f}_h(x)$ on predicting their commonly-shared image category c_j.

This loss function in Eq.(4) is used to learn τ base networks and a τ-D gate network jointly for generating the network $\mathbb{F}(x)$, and it has three parts:

(a) The ***first part*** aims to minimize the loss of the relevant base deep network.

(b) The ***second part*** is the gate network loss to emphasize that: (1) for the same training image $\boldsymbol{x}_j^l$, its best-matched category c_j can be identified correctly by multiple base network (c_j is assigned into multiple task groups because of inter-group task overlapping), thus the corresponding base deep network should have good consistency on their confidence scores on predicting their commonly-shared image category c_j; (2) for τ base deep networks, their individual contributions on the mixture network largely depend on their confidences. Thus the τ-D gate network is learned to depress the irrelevant base deep networks (i.e., the task groups do not contain the corresponding image category) and promote the relevant base deep networks (i.e., the task groups contain the corresponding image category). In the training time, when the corresponding image category for a given image belongs to the current base deep network, its confidence should be promoted, otherwise, it should be treated as irrelevant base deep network and its confidence should be depressed. Thus such confidence scores $\boldsymbol{\phi}_t$ can be treated as a good factor to decide which deep networks are more reliable and have more contributions on the mixture network $\mathbb{F}$.

(c) The ***third part*** aims to address the overconfidence issue and guarantees that the best-matched category for each image has higher probability than others in a reasonable margin β, e.g., $P_{opt}(\boldsymbol{x}_j^l, y_j^l) - P_{opt}(\boldsymbol{x}_j^l, c_j) \geq \beta$. The idea behind our hinge loss is to depress the irrelevant base deep networks when they make wrong predictions with high probability. Such confident hinge loss can guarantee that each base deep network is trained to recognize different subsets of 10,184 image categories in ImageNet10K.

4 Experimental Results

To evaluate the performance of our deep mixture model (DeepME, for short), a series of experiments are conducted on the chosen image classification benchmark dataset, which is further compared to several state-of-the-art peer baselines.

4.1 Dataset

We performed experiments on ImageNet10K, which is one of the most well-known image datasets for visual classification, and contains 10,184 image categories and 9M images. Furthermore, we use a 85%-5%-10% train/validation/test split. In all experiments, we compute the top-1, top-3 and top-5 accuracy per class and the average accuracy, which could well evaluate the performance on image classification.

Besides, we provide details of the training process to exhibit the experimental procedure more in-depth. In DeepME, the training process consists of two parts as follows:

1. **Training of the base CNN**. We utilize Stochastic Gradient Descent (SGD) with momentum 0.9 to learn the base network. The training is divided into two stages: 1) The warm-up stage and 2) The fine-tuning stage. In the warm-up stage, the learning rate is set from 0.01 to 0.001 in a exponentially decayed manner, while in the fine-tuning stage it is set from 0.001 to 0.00001. We use batch size 256 and L2 regularization for the corresponding parameters with weight 0.0005.
2. **Training of the gate network**. To initialized the parameters of the gate network, *GlorotNormal* initializer is adopted as suggested in [Orr and Müller, 2003]. SGD with momentum is also used to learn the network, and the batch size is 256. The initial learning rate is 0.001 and then exponentially decayed to 0.0001.

4.2 Baselines

We compare our deep mixture method (DeepME) with several state-of-the-art methods and comparing experiments aim to exploring the effectiveness of our proposed model. Thus the baselines are used in our experiments as follows:

- **VGG16-Ext**. This is a straightforward but common manner to utilize a pre-trained model to perform a new task. The fully connected layer of VGG16 is enlarged from a 1,000-way to a 10,184-way softmax. The training of VGG16-Ext is similar to the base model of our method, i.e., first update the last layer to warm up, and then fine-tune all layers until convergence.
- **DMDE**. As DMDE [Zhao *et al.*, 2018] has achieved the best performance based on mixture of experts so far, we can directly know the performance of our proposed DeepME model by comparing it with DMDE. For the sake of fairness, we also replace the base network (i.e., AlexNet) of DMDE with VGG16.
- **DeepME-NoSSC**. To better evaluate the significance of Soft Spectral Clustering in DeepME, we devise a baseline named as DeepME-SSC without SSC, i.e. we obtain the 15 clusters by random.
- **DeepME-NoGate**. To evaluate the significance of the gate network, we devise a baseline named as DeepME-NoGate without the gate network. In this setting, all the expert networks (VGG16 in our case) contribute equally.

4.3 Overall Performance

First of all, to demonstrate the effect of our model, we present the comparisons of average accuracy on top 1, top 3 and top 5 respectively, which is exhibited in Table 1. The results clearly indicate that DeepME has achieved the best performance among all baselines. Particularly, our model (32.13%) moves a big step forward compared with DMDE (29.51%), which is the best model based on mixture of experts so far. This significant improvement roughly validates the effectiveness of the soft clustering model (i.e., SSC) and the sophisticated gate network. Particularly, compared with VGG16-Ext,

Model	Top 1	Top 3	Top 5
VGG16-Ext	27.62%	45.71%	53.17%
DMDE	29.51%	48.75%	56.49%
DeepME-NoSSC	20.24%	40.16%	49.20%
DeepME-NoGate	7.84%	21.36%	29.85%
DeepME	**32.13%**	**51.72%**	**59.79%**

Table 1: Average accuracy on different models.

the performance (27.62%) in ImageNet10K is not satisfactory, which validates the intuition "simply enlarging the final outputs of some well-designed deep CNNs may not be an optimal solution for large-scale image classification".

To explore the relative importance of SSC and the gate network, we conduct a analysis on the results of DeepME-NoSSC (20.24%) and DeepME-NoGate (7.84%). Based on the results, we could draw several conclusions: a) Both SSC and the gate network play important roles in classifying large-scale images. If SSC is removed from DeepME, the accuracy will decrease from 32.13% to 20.24%. Similarly, when gate network is absent, the accuracy will dramatically decrease from 32.13% to 7.84%, which indicates that simple fusion functions hardly achieve acceptable results; b) Comparing with SSC, the importance of the gate network (DeepME-NoGate: 7.84%) is far greater than SSC (DeepME-NoSSC: 20.24%). We conjecture that the reason of the phenomenon is the strong learning ability of the gate network. In a word, our deep mixture algorithm with SSC and the gate network performs well in large-scale image classification.

We did not back-propagate the gradients through the gate network to fine-tune the importance of the base networks subjected to limited computing resources. However, the performance of DeepME without fine-tuning surpasses all baselines. Theoretically, the performance of DeepME with fine-tuning will be superior than DeepME without fine-tuning.

4.4 Specific Performance

Table 1 has exhibited the overall performance (i.e., average accuracy) of all methods. To better understand the effectiveness of DeepME, in this section, we provide DMDE and DeepME's accuracy distributions (top 5) on each 10184 image class in an ascending order as shown in Fig. 3. Please note that we only provide DMDE's results as the comparison for simplicity, which has achieved the best performance on baselines. Similar to Section 4.3, Fig. 3 also well demonstrates the classifying superiority of DeepME compared with DMDE. Especially, on some image classes, DMDE is hard to accurately identify these classes (e.g., *Dialyzer*: 10.34%, *Shrimpfish*: 14.29%, *Seismograph*: 15.79% and so on), while our proposed model DeepME could accurately classify these hard image classes (for example, *Dialyzer*: 72.41%, *Shrimpfish*: 74.29%, *Seismograph*: 68.42% and so on).

4.5 Hard Task vs. Easy Task

According to the designed SSC, the large-scale classification task is divided into 15 sub-tasks. During the training of each expert network, we find that there are significant differences in complexity among tasks. To be specific, the average accuracy of these 15 tasks is 45.7%, while the best one (*Task 1*) is 55.4% and the worst one (*Task 2*) is only 31.2%. It implies that DeepME could effectively generate sub-tasks based on the learning complexity. As a matter of fact, *Task 2* is indeed harder than *Task 1*. The former (i.e., *Task 2*) focuses on classifying trees (especially, various types of coniferous trees), which are very similar to each other. However, the latter (i.e., *Task 1*) is focusing on classifying sports, which are discrepancy to each other. In practice, if we want to improve the performance of large-scale image classification notably, we should pay more attention on the hard tasks (e.g., *Task 2*), which have more room for improvement.

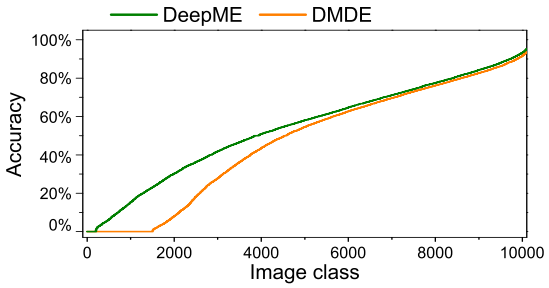

Figure 3: The specific accuracy on each 10184 class.

5 Conclusion

In this paper, to support large-scale image classification (i.e., classifying images into tens of thousands of classes), an adaptively deep mixture method (DeepME) was developed to support large-scale image classification. Three main contributions could be concluded as follows: (a) a soft spectral clustering method was developed to assign large numbers of image categories into a set of groups according to their inter-category semantic correlations; (b) a two-layer ontology (group layer and class layer) was constructed to organize large numbers of image categories hierarchically, which was further used to assign the semantically-similar image categories with similar learning complexities into the same task group; (c) a gate network was learned to determine the importances for all these base deep networks automatically and combine their fewer diverse outputs adaptively to generate a mixture network with much larger outputs (i.e., 10,184-D outputs for handling ImageNet10K set with 10,184 image categories). Finally, we conducted extensive experiments on ImageNet10K image data set, and our deep mixture algorithm achieved very competitive performance compared with several baselines. In the future, we will further investigate whether the gate network and the base CNNs could be trained in an end-to-end fashion. In addition, more interesting fusion methods, e.g., attention mechanism, are also worth studying.

Acknowledgments

This research was partially supported by grants from the National Key Research and Development Program of China (No. 2018YFC0832101), and the National Natural Science Foundation of China (No. 61922073, 61631014).

References

[Bach and Jordan, 2004] Francis R Bach and Michael I Jordan. Learning spectral clustering. In *Advances in neural information processing systems*, pages 305–312, 2004.

[Bezdek *et al.*, 1984] James C Bezdek, Robert Ehrlich, and William Full. Fcm: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3):191–203, 1984.

[Borisyuk *et al.*, 2018] Fedor Borisyuk, Albert Gordo, and Viswanath Sivakumar. Rosetta: Large scale system for text detection and recognition in images. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 71–79, 2018.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[Ge *et al.*, 2016] ZongYuan Ge, Alex Bewley, Christopher McCool, Peter Corke, Ben Upcroft, and Conrad Sanderson. Fine-grained classification via mixture of deep convolutional neural networks. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–6. IEEE, 2016.

[Hinton *et al.*, 2006] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[Howard *et al.*, 2019] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, et al. Searching for mobilenetv3. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1314–1324, 2019.

[Jacobs *et al.*, 1991] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[Jarrett *et al.*, 2009] Kevin Jarrett, Koray Kavukcuoglu, Yann LeCun, et al. What is the best multi-stage architecture for object recognition? In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2146–2153. IEEE, 2009.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[Leacock and Chodorow, 1998] Claudia Leacock and Martin Chodorow. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283, 1998.

[LeCun *et al.*, 1998] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[Li *et al.*, 2020] Zhi Li, Bo Wu, Qi Liu, Likang Wu, Hongke Zhao, and Tao Mei. Learning the compositional visual coherence for complementary recommendations. *arXiv preprint arXiv:2006.04380*, 2020.

[Ma *et al.*, 2018] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1930–1939, 2018.

[Ma *et al.*, 2020] Benteng Ma, Xiang Li, Yong Xia, and Yanning Zhang. Autonomous deep learning: a genetic dcnn designer for image classification. *Neurocomputing*, 379:152–161, 2020.

[Masoudnia and Ebrahimpour, 2014] Saeed Masoudnia and Reza Ebrahimpour. Mixture of experts: a literature survey. *Artificial Intelligence Review*, pages 1–19, 2014.

[Miller, 1995] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[Nguyen *et al.*, 2019] Hien D Nguyen, Faicel Chamroukhi, and Florence Forbes. Approximation results regarding the multiple-output gaussian gated mixture of linear experts model. *Neurocomputing*, 366:208–214, 2019.

[Orr and Müller, 2003] Genevieve B Orr and Klaus-Robert Müller. *Neural networks: tricks of the trade*. Springer, 2003.

[Rao *et al.*, 2018] Jinfeng Rao, Ferhan Ture, and Jimmy Lin. Multi-task learning with neural networks for voice query understanding on an entertainment platform. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 636–645, 2018.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[Sun *et al.*, 2014] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *Advances in neural information processing systems*, pages 1988–1996, 2014.

[Sun *et al.*, 2019] Yanan Sun, Bing Xue, Mengjie Zhang, and Gary G Yen. Evolving deep convolutional neural networks for image classification. *IEEE Transactions on Evolutionary Computation*, 2019.

[Tang *et al.*, 2012] Yichuan Tang, Ruslan Salakhutdinov, and Geoffrey Hinton. Deep mixtures of factor analysers. *arXiv preprint arXiv:1206.4635*, 2012.

[Wang *et al.*, 2020] Xin Wang, Wei Huang, Qi Liu, Yu Yin, Zhenya Huang, Le Wu, Jianhui Ma, and Xue Wang. Fine-grained similarity measurement between educational videos and exercises. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 331–339, 2020.

[Zhang *et al.*, 2019a] Kai Zhang, Hefu Zhang, Qi Liu, Hongke Zhao, Hengshu Zhu, and Enhong Chen. Interactive attention transfer network for cross-domain sentiment classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5773–5780, 2019.

[Zhang *et al.*, 2019b] Lianbo Zhang, Shaoli Huang, Wei Liu, and Dacheng Tao. Learning a mixture of granularity-specific experts for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8331–8340, 2019.

[Zhang *et al.*, 2020] Haixi Zhang, Zhenzhong Kuang, Xianlin Peng, Guiqing He, Jinye Peng, and Jianping Fan. Aggregating diverse deep attention networks for large-scale plant species identification. *Neurocomputing*, 378:283–294, 2020.

[Zhao *et al.*, 2018] Tianyi Zhao, Qiuyu Chen, Zhenzhong Kuang, Jun Yu, Wei Zhang, Ming He, and Jianping Fan. Deep mixture of diverse experts for large-scale visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[Zhu *et al.*, 2020] Hongyuan Zhu, Qi Liu, Nicholas Jing Yuan, Kun Zhang, Guang Zhou, and Enhong Chen. Pop music generation: From melody to multi-style arrangement. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14(5):1–31, 2020.