IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 9 April 2024, accepted 4 May 2024, date of publication 8 May 2024, date of current version 17 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3398134

RESEARCH ARTICLE

DCPNet: Distribution Calibration Prototypical Network for Few-Shot Image Classification

RANHUI XU, KAIZHONG JIANG, LULU QI, SHAOJIE ZHAO, AND MINGMING ZHENG
School of Mathematics, Physics and Statistics, Shanghai University of Engineering Science, Shanghai 201620, China

Corresponding author: Kaizhong Jiang (m440121104@sues.edu.cn)

This work was supported by the National Statistical Science Research Project of China under Grant 2020LY080.

ABSTRACT Deep learning has witnessed significant advancements in various tasks and has displayed exceptional performance. However, traditional deep learning techniques often necessitate the utilization of extensive labeled data for training, a requirement that is challenging to fulfill in many real-world scenarios. This limitation has given rise to the field of few-shot learning (FSL). In this paper, we introduce a Distribution Calibration Prototypical Network (DCPNet), aiming to address the limitations of prototypical networks in terms of their weak feature extraction capabilities and the inability of their classifier boundaries to align with the dataset. DCPNet incorporates a parallel hierarchical feature extraction module and a few-shot differentiation loss function to fine-tune the metric learning for better feature representation. This approach employs a parallel approach to extract features based on the semantic depth of image hierarchical extraction and incorporates contrastive learning to achieve feature vector fusion. Furthermore, DCPNet incorporates an improved distribution calibration method that leverages information from the base class dataset to align classifier boundaries with the dataset. To validate our approach, we conducted comparative experiments on datasets such as Mini-Imagenet, Omniglot, and CUB using classical baseline methods. In additional, we conducted ablation experiments on the Mini-Imagenet to assess the performance effectiveness of each component of the model. The results demonstrate that the proposed method presented in this paper outperforms other approaches and offer new insights into the field of few-shot image classification.

INDEX TERMS Improved distribution calibration, few-shot learning, prototypical network, image classification, computer vision.

I. INTRODUCTION

Image classification is a fundamental task of computer vision and is of great significance for automatic driving, intelligent security, and medical image analysis, which refers to the process of identifying the category to which a given image belongs. Traditional neural networks require a large amount of labeled image data to solve image classification problems, but a large amount of labeled data is not available in practical situations; therefore, the study of few-shot learning methods is particularly important. To address the scenario of traditional image classification methods with a small amount of labeled sample data, scholars have proposed the few-shot image classification methods.

To align with the intricacies of few-shot learning, scholars have categorized their proposed research methods into three distinct categories: those rooted in metric learning, optimization-based learning, and data augmentation techniques. This paper presents an improvement based on the prototypical network in metric learning. This network is made by extracting the image data into feature vectors and then clustering them into prototype centers. Then, it can calculate and compare the distance between the query samples and prototype centers for category discrimination. However, based on the design of the prototypical networks, we observe two important factors that affect their classification performance. One is that the embedding learning of the prototypical network is based on the distribution of support vectors across the class, which may not capture the subtle relationships between support vectors, such as their correlations. The

The associate editor coordinating the review of this manuscript and approving it for publication was Massimo Cafaro.

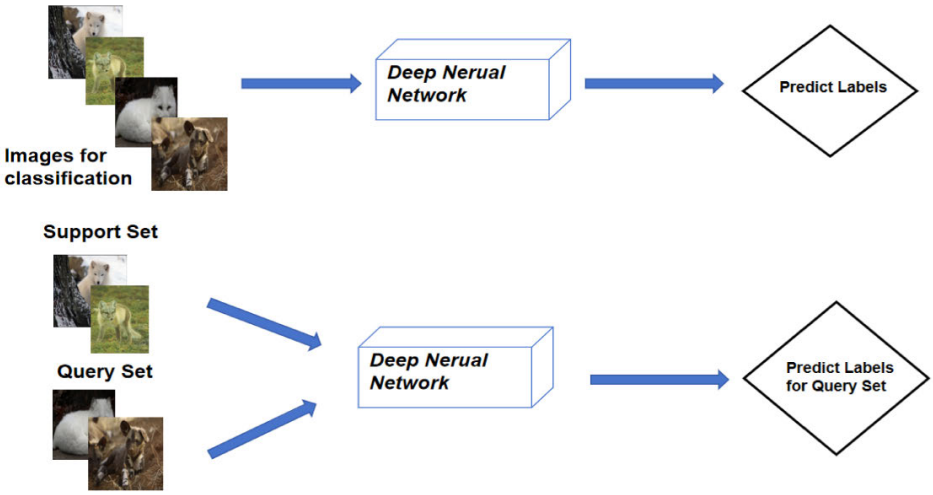

FIGURE 1. The difference of images classification and few-shot classification.

second factor is that the self-learning capabilities of the model are easily disrupted by errors in the measurement method and feature extraction. The distribution of samples across classes can affect the performance of the prototypical networks. The performance of prototypical networks can deteriorate if the distributions of samples from different classes are too similar. We suggest implementing a parallel hierarchical feature extraction module and a few-shot differentiation loss function to address the limitations of incompleteness and lack of detail in feature extraction. Additionally, we proposed an improved distributional calibration method to incorporate prior knowledge into the classifier to make it more appropriate for a given dataset to address the limitations of the classifiers.

- Our method addresses the problems of weak feature extraction and incomplete extracted information in prototype networks by proposing a parallel hierarchical feature extraction module and few-shot differentiation loss function.
- Our approach proposes an improved distribution calibration for problems in which prototype network classifiers have rigid boundaries and cannot be better adapted to new datasets.
- To verify the experimental performance of our method, we conducted experiments on three public datasets in comparison with some baselines and the results show that our method can be roughly optimal compared with other methods, and only one dataset fails to reach the optimally, but the results are similar.

The remainder of this paper is organized follows. Section II presents related work on image classification. Section III describes the proposed DCPNet method. Section IV presents the experimental results. Finally, section V provides a summary of this study.

II. RELATED WORK

A. FEW-SHOT LEARNING

Machine learning and deep learning excel in scenarios with abundant labeled data and deliver remarkable performance. Nevertheless, as technology and society progress, numerous real-life situations struggle to generate substantial labeled sample data, thereby posing significant challenges for model training. The emergence of few-shot learning has considerably mitigated this issue. According to the definition of the few-shot problem provided by Wang et al. [1] in 2020, a computer program segment can learn from experience E and enhance its performance P on a given task T, considering a specified performance metric P. In the context of few-shot learning, experience E is acquired through a limited number of supervised samples related to task T, thereby facilitating learning in data-scarce environments.

The initial strategy aims to improve sample diversity in few-shot learning scenarios by utilizing data augmentation, given the limited size of the available samples. Data augmentation primarily encompass three categories: those based on unlabeled data, those based on data synthesis, and those based on feature enhancement. Notable examples include MEDA [2], Label Hallucination [3] and other techniques for few-shot classification. In a study by Wang et al. [4], virtual data were generated to enhance the diversity of samples using a data generation model. This approach combined with meta-learning allowed for the training of generative models and classification algorithms in an end-to-end manner, resulting in promising outcomes. Xian et al. [5] proposed a unified learning framework for inductive and transductive feature generation in any-shot learning. They introduced a conditional generative model that combined the strengths of VAE [6] and GAN [7]. In addition, the authors utilized an unconditional discriminator to learn the edge feature distributions of the unlabeled images. This approach offers a versatile framework for feature generation in a few-shot learning settings.

The second methodological approach for addressing the few-shot problem is rooted in optimisation-based learning. This approach defines a general optimization technique that can be applied to all models, optimizing all potential classes

using the algorithm, rather than solely optimizing for a specific dataset. Prominent examples include MAML [8], RelationNet [9], CovaMNet [10], ANIL [11], Free lunch [12], and others, which train models using meta-training datasets composed of multiple small tasks to facilitate rapid adaptation to new tasks. Zhang et al. [13] presented MRA-GNN, a novel network that utilizes multi-granularity graphs to achieve few-shot learning with improved generalization ability. Furthermore, there are algorithm-based few-shot learning algorithms, which are based on prior knowledge to modify the search for optimal hypotheses within a given hypothesis space. Zhang et al. [14] developed a few-shot image classification method from a novel perspective, utilizing optimal matching between image regions. To adopt this approach, they designed a cross-reference mechanism that effectively mitigates the adverse effects of background clutter and large intra-class appearance variations. So the method of using Earth Mover's Distance to calculate the similarity between images is proposed. Wang et al. [15] proposed a selective attack module that consists of trainable adapters. Each adapter generates a spatial attention map of the image used to guide attacks on category-independent image regions to capture key features and correct the visual distribution of image features. Secondly, they also utilised Earth Mover's Distance in order to optimise the prototype and derive an upper bound for the Earth Mover's Distance. Again, they also used an augmentation strategy to prevent overfitting in few-shot learning. In few-shot learning, tasks in the meta-training dataset can involve learning new categories with limited samples. After the meta-training phase, the model can quickly adapt to novel few-shot tasks. Finn et al. [8] developed MAML, a meta-learning-based few-shot learning strategy that optimizes the model using a meta-parameter and fine-tunes it for each task, giving the model superior generalization ability. Another few-shot learning approach involves graph neural networks, which leverage the power of graph representation learning for data such as images and texts. This method has shown strong performance in few-shot learning. Dai et al. [16] proposed PFEMed, which is the extraction of general and special features from medical images using a dual encoder structure, in addition to a priori guided auto-variance encoder that is used to enhance the robustness of the target features. The target query set features and support set features are then matched to select for query set category prediction.

The final category encompasses the methods that rely on metric learning. These techniques leverage prior knowledge of high similarities within the same category and between different categories to simplify complex models into ones that can measure the similarity between different samples. Notable examples include Matching Networks [17] and Prototypical Networks [18] and others. Prototypical Networks [18] translates the support samples of each class into a D-dimensional embedding space and computed the prototype embedding for each class. Then, by computing the similarity between the query embedding and each prototype embedding using Euclidean distance, each query sample is assigned the class with the most similar prototype embedding. To obtain the prototype embedding for a class, the prototype networks compute the mean of the support embedding vectors within that class in the support set. Gao et al. [19] identified the challenges associated with obtaining stable large-scale supervised training datasets, noting that existing methods for relationship classification primarily rely on distant supervision. As few-shot learning typically focuses on low-noise images, it can be challenging to directly handle diverse text information. They introduce a hybrid attention-based prototype network to address the noisy few-shot relation classification problem. This prototype network incorporates instance-level and feature-level attention mechanisms to highlight important instances and features, respectively. Fort [20] proposed the integration of a Gaussian process into a prototype network. In this approach, each image is mapped to an embedding vector and an estimate of image quality. A Gaussian covariance matrix is then used to predict and characterize a confidence interval. Liu et al. [21] improved the loss function by considering the balance between the discriminative and migratory aspects of the few-shot model. They introduced a margin-based softmax loss to enhance performance. Nguyen et al. [22] introduced SEN, an improvement in Euclidean distance, which eliminates the need for significant normalization but still achieves normalization. A novel variational inference network called TRIDENT was introduced by Singh et al. [23]. They separated images into latent variables for image semantics and labels, inferring them in an alternating manner. They employed an internal attention-based feature extraction module called AttFEX to foster task perception, effectively utilizing information from both query and support images. Hu et al. [24] conducted extensive research on few-shot learning, focusing on dataset, architecture, and fine-tuning strategy. The experimental results highlighted the significance of the source dataset and neural network structure on the few-shot learning performance. Furthermore, data enhancement for fine-tuning the feature backbone is essential when domain divergence occurs between the training and testing sets. Ma et al. [25] studied the few-shot classification problem from a geometric perspective and they found that the essence of a prototype network can be regarded as a Voronoi Diagram in the feature space. Based on this perspective, they proposed Cluster-induced Voronoi Diagram to improve the accuracy and robustness of the few-shot image classification. SetFeat [26] was proposed by Afrasiyabi et al. They constructed an image representation of the base class in terms of a set representation and used a shallow attention mechanism to improve model accuracy. Hiller et al. [27] used Vision Transformer to establish region semantic relationships in images and also provide interpretability of images, learning a more general statistical structure of the data to overcome supervised collapse.

B. CONTRASTIVE LEARNING

Contrastive learning [28] is a supervised learning approach that compares data points to understand similarities and differences within a given model. Unlike traditional classifier learning, contrastive learning considers not only the similarity within the same class but also the dissimilarity between different classes. To train the model, contrastive learning primarily focuses on optimizing the distance between encoder vectors. By bringing similar samples closer together and negative samples further apart, the model could learn more effectively. Furthermore, by reducing the reliance on labeled data, contrastive learning enables accurate data representation even without explicit labels. Currently, several contrastive learning methods are available. These methods can be broadly categorized into negative example-based contrastive learning, contrastive clustering, asymmetric network structures, and methods based on redundancy removal loss functions to prevent overfitting. SimCLR [29] is a negative example-based contrastive learning framework that does not require any special architecture or memory banks. Li et al. [30] proposed a clustering algorithm based on contrastive learning that can simultaneously perform representation learning and clustering analysis, making it suitable for streaming data clustering. The BYOL approach, introduced by Grill et al. [31], uses an asymmetric network structure and achieves excellent classification accuracy on datasets like ImageNet [32] that lack negative samples. This distinguishes it from MoCo [33]. In addition, Zbontar et al. [34] introduced a loss function based on redundancy elimination, effectively preventing model collapse and ensuring the appropriate design of the loss function. Liang et al. [35] designed a new hierarchical contrast learning strategy to capture the correlation and difference between target-invariant and feature-specific features and used it to achieve good performance in few-shot image classification task in an up-zero-shot scenario.

In the research of Schroff et al. [36], triplet loss was proposed in the field of contrastive learning, the basic idea is that the samples with the same labels and their feature vectors are as close as possible to each other in the embedding space and the samples with different labels and the distance of their feature vectors are as far as possible from each other in the embedding space. First, the samples are classified into three categories: anchor, positive and negative samples, where anchor means randomly selected samples, positive means samples with the same category as the anchor samples, and negative means samples with different categories from the anchor. Finally, the expression for the calculation is entered, the expression is as follows:

$$Loss = max\left(d\left(a, p\right) - d\left(a, n\right) + margin, 0\right) \quad (1)$$

where, a denotes a random sample in a category randomly selected from among the data categories is called an anchored sample. p denotes a randomly selected sample in the same category as the anchored sample a that different from anchored sample is called positive sample. n denotes a randomly selected sample in a category different from the anchored sample is called negative sample. $d(x, y)$ denotes distance function, often used as a euclidean distance function. The goal of minimizing the loss is that $d(a, p)$ is close to 0 and $d(a, n)$ is greater than $d(a, p) + margin$, where the margin denotes an artificially set constant greater than 0.

III. METHODOLOGY

A. PROBLEM DEFINITION

We followed a typical few-shot image classification setting. Given a labeled dataset $D = (x_i, y_i)$ where $x_i \in R^n$ is the feature vector of images sample and $y_i \in C$ where C denotes the set of label classes. We can spilt the label into the base classes C_{base} and novel classes C_{novel}.Where $C_{base} \cap C_{novel} = \oslash$ and $C_{base} \cup C_{novel} = C$. Our approach is to train a model based on the data, which can generalize from the basic classes to the novel classes; in fact, we can sample few-shot tasks randomly. The most common way to build a task called N-way K-shot task, which means N classes are randomly sampled from the novel set and K such as 1 and 5 labeled samples from each class in one task, in which the few-shot available data can split two sets called support set and query set. The support set contains kn samples and the query set contains kq samples. Thus, the performance of the model was evaluated as the average accuracy on tasks randomly sampled from the novel classes.

B. OVERVIEW OF METHOD

Next, we delve into the parallel hierarchical feature extraction module and the few-shot differentiation loss function. These components demonstrate how the model prototype center's representatives can be refined. Subsequently, we elaborate on how the improved distribution calibration can fine-tune the model classification surface, ensuring that it is better suited to the new dataset.

During the training phase, we meticulously organized the data into triplet and few-shot support set forms and subsequently inputted them into the parallel hierarchical feature extraction module. This process yields the feature vector, which is we then used to calculate both the triplet loss and cross-entropy loss in comparison with the ground truths. These losses are weighted to formulate the few-shot differentiation loss, facilitating back-propagation and model training. Our innovative parallel hierarchical feature extraction module, along with the few-shot differentiation loss function, represents an advanced metric-learning approach rooted in prototype networks. By harnessing deep and shallow semantic separation, splicing techniques, and contrastive learning, we cultivate a more nuanced metric learning model that adeptly distances dissimilar samples while drawing similar ones closer together. Once the loss value stabilizes, we transition into the testing phase. Here, we employ an improved distribution calibration strategy to refine the classification boundaries and evaluate the classifier performance through rigorous testing.

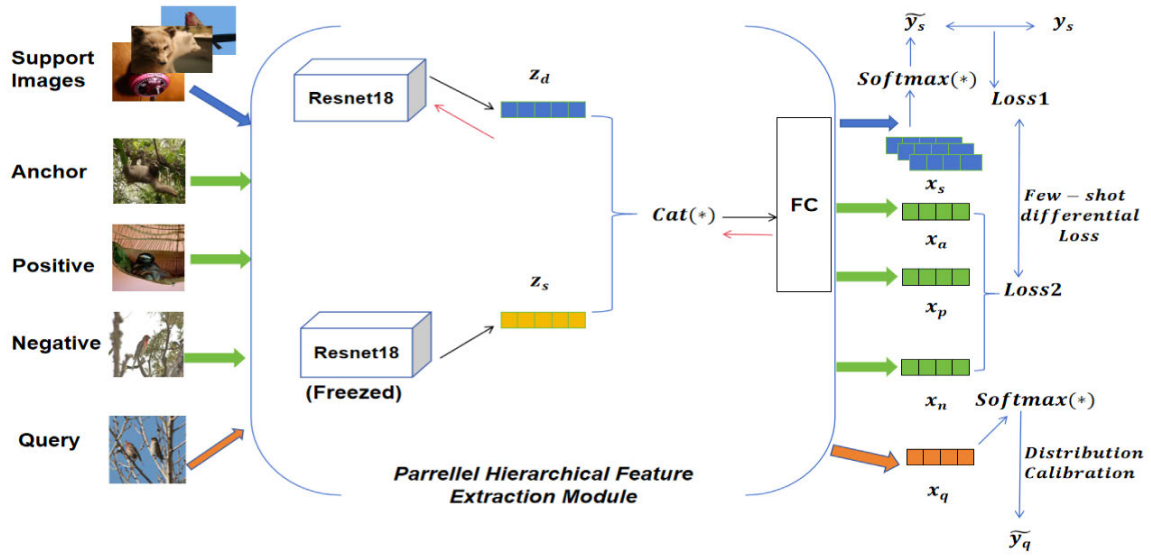

FIGURE 2. DCPNet: the structure of DCPNet method.

C. PARALLEL HIERARCHICAL FEATURE EXTRACTION MODULE

As we introduced the problem definition before, the support and query images are first converted into feature vectors. The proposed approach employs a two-branch encoder structure. We transferred the most common Resnet18 given the simplicity and lightness of the model. We designed two branches as the feature extraction encoders, one branch followed the model for training, maintaining gradient propagation to extract specific deep semantic image features. The other branch freezes the gradient and does not follow the model for training to extract general shallow semantic features for the model. Subsequently, the final output of the feature extraction encoder is used by merging two features and resizing them to the appropriate size using a Fully-Connected layer.

Suppose that a single image data is x_i, which is $f_1(x_i)$ after the first branch encoder and $f_2(x_i)$ after the other encoder. Then, the feature extracted from this single image data can be represented as:

$$f(x_i) = FullyConnect(Cat(f_1(x_i), f_2(x_i))) \tag{2}$$

where, $Cat(x, y)$ denotes tensor affine operator and (x, y) denotes a pair of tensors. This can be done with torch.cat in Python.

D. FEW-SHOT DIFFERENTIATION LOSS

In this subsection, we introduce the proposed few-shot differentiation loss function to address the problem of inaccurate representation of prototype networks prototypes. We first divide the implementation process of completing the task into four stages: Feature extraction, Prototype representation, Distribution calibration, Completion of classification. In feature extraction we use a two-brunch approach to get the initial prototype vectors, in order to make the model more portable, we abandoned the use of deeper and more complex neural networks and directly migrated the Resnet18.

In the prototype representation stage, we are still analogous to prototype networks for prototype-centred representation learning. However, to obtain a better prototype representation, we use the triplet loss function of contrastive learning and weight it with the cross-entropy loss function to get the final loss function:

$$TotalLoss = \lambda \left\{ max \left[d\left(x_i^a, x_i^p\right) - d\left(x_i^a, x_i^n\right) + margin \right], 0 \right\} \tag{3}$$

$$+ (1 - \lambda) \left[- \sum_i p_i log\left(q_i\right) \right] \tag{4}$$

where x_i^a denotes the anchor sample selected in the support set, x_i^p denotes the positive sample selected in the support set with the same category as the anchor sample, x_i^n denotes the negative sample selected from the support set with a different category than the anchor sample, margin denotes the artificially set hyper-parameter controlling the difference in the distance between the samples. In the cross-entropy loss function, p_i denotes the true value and q_i denotes the predicted value. λ denotes the weighted weight that controls the two sets of loss functions.

Our starting point is to increase the distance between dissimilar samples and decrease the distance of similar samples in the triplet loss of contrastive learning, so as to get a prototype representation that is easier to classify.

E. IMPROVED DISTRIBUTION CALIBRATION

First, we assumed that the feature vectors obtained by the encoder followed a Gaussian distribution. Statistical calculations were performed based on the data of the base categories and the mean and variance of each category were

calculated, where the mean and covariance are expressed as follows:

$$\mu_i = \frac{1}{n_i}\sum_{j=1}^{n_i} x_j \tag{5}$$

$$\sum_i = \frac{\sum_{j=1}^{n_i}(x_j - \mu_i)(x_j - \mu_i)^T}{n_i - 1} \tag{6}$$

As our statistics are based on the sample estimation of the whole, the sample covariance is used instead of the overall covariance.

Similar to in the method proposed by Shuo Yang et al [12], we used the same training technique for the prototype network. Based on previous research, we directly transferred the feature vectors acquired after WideResNet [37] training and computed the statistics based on the feature vectors. and then transferred these statistics to the prototype representation of the novel class after the feature extraction module. Similarly, we select the k base classes with the support set chosen to have the closest Euclidean distance to the sample features. However, the difference is that we do not perform a Turkey's Ladder of Powers transformation of the sample features, but only data normalization pre-processing.

$$\mathbb{S}_d = \left\{ -\|\mu_i - \tilde{x}\|^2 \mid i \in C_{base} \right\} \tag{7}$$

$$\mathbb{S}_N = \left\{ i \mid -\|\mu_i - \tilde{x}\|^2 \in topk\left(\mathbb{S}_d\right) \right\} \tag{8}$$

where $topk\left(\right)$ is the closest top k elements from input set, which means the feature vector $\tilde{x}$. We select the closest k base classes to compute the corrected mean and variance.

$$\mu' = \frac{\sum_{i\in\mathbb{S}_N}\mu_i + \tilde{x}}{k+1} \tag{9}$$

$$\Sigma' = \frac{\sum_{i\in\mathbb{S}_N}\Sigma_i}{k} + \alpha \tag{10}$$

We use constructed corrected statistics to construct a new Gaussian distribution and sample it.

$$\mathbb{D}_y = \left\{ (x, y) \mid x \sim \mathcal{N}\left(\mu, \Sigma\right),\ \forall\left(\mu, \Sigma\right) \in \mathbb{S}_y \right\} \tag{11}$$

where $\mathbb{D}_y$ denotes the set of corrected feature vectors constructed from feature vectors sampled from the new Gaussian distribution. Where $\mathbb{S}_y$ denotes the set of mean and variances after k corrections. Generate new feature vectors to be fused with the feature vectors of the query set and train a logistic regression classifier.

IV. EXPERIMENTS

Our proposed method, DCPNet, was experimentally evaluated against classical few-shot classification methods on three publicly available datasets. The experimental results revealed that DCPNet outperformed classical methods in terms of both learning performance and time efficiency. The following seven subsections provide relevant details regarding the datasets, experimental settings, implementation details and results of our experiments.

Algorithm 1 Training a DCPNet

Data: Training data $= \{(x_1, y_1), \ldots, (x_N, y_N)\}$;
N_c: The number of classes per episode;
K: The number of classes in the training set;
N_s: The number of support examples per classes;
N_Q: The number of query examples per classes
Result: $\hat{y}$
Fix *TotalLoss*: Total loss of DCPNet;
$f\ ()$: The encoder model of DCPNet;
Choose a controlled hyper-parameter λ;
$V \longleftarrow RandomSample\left(\{1..K\}, N_c\right)$;
$Totalloss = 0$;
for $k \in V$ **do**
  $X_i^a, X_i^p, X_i^n \longleftarrow TripletRandomSample\left(D\left(x, y\right)\right)$;
  $f\left(x_i^a\right), f\left(x_i^p\right), f\left(x_i^n\right) = model\left(X_i^a, X_i^p, X_i^n\right)$;
  $S_k \longleftarrow RandomSample\left(D_{vk}, N_S\right)$;
  $Q_k \longleftarrow RandomSample\left(D_{vk} \backslash S_k, N_Q\right)$;
  for $(x_i, y_i) \in Q_k$ **do**
    $C_k \leftarrow \frac{1}{N_c}\sum_{(x_i,y_i)\in S_k} f\left(x_i\right)$;
    $TotalLoss =$
     $\lambda * TripletLoss\left(f\left(x_i^a\right), f\left(x_i^p\right), f\left(x_i^n\right)\right) +$
     $(1-\lambda) * CrossEntropy\left(y_i, C_k\right)$;
    $TotalLoss = TotalLoss - \frac{\partial^2 TotalLoss}{\partial x_i^a \partial x_i} TotalLoss$;
  end for
end for
$V' \longleftarrow RandomSample\left(\{1..K\}, N_c\right)$;
for $k \in V'$ **do**
  for $(x, y) \in Q_k$ **do**
    $(\tilde{x}, y) \leftarrow DistributionCalibration\left(x, y\right)$;
    $\hat{y} \leftarrow LogitClassifier\left(\tilde{x}\right)$;
  end for
end for

A. DATASETS

For our experiments, we chose three datasets: including Mini-Imagenet [17], Omniglot [38] and CUB [39]. Mini-Imagenet [17] is a widely used dataset for few-shot learning. This dataset consists of 60,000 colorful images divided into 100 categories, each of size 84 × 84 pixels. Eighty percent of the dataset comprises the training set, while the remaining part is the test set. Another dataset we chose for the experiments is Omniglot [38], which is also widely used in few-shot learning scenarios. This dataset comprises 50 different language alphabets. Each alphabet contains 1,623 distinct characters, and every character is written by 20 different individuals. Each image in this dataset has a size of 105 × 105 pixels. The last dataset we chose for our experiments is CUB (Caltech-UCSD Birds-200) [39], which is a widely used benchmark image dataset for few-shot fine-grained classification and recognition research. This dataset comprises 11,788 bird images, categorized into 200 different bird subcategories. Out of the total dataset, 5,994 images

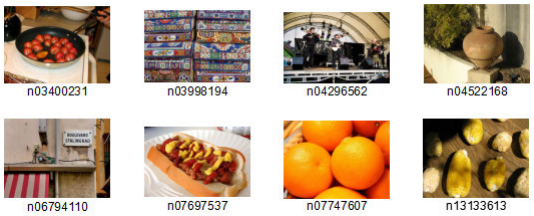

FIGURE 3. Mini-Imagenet: Partial images and labels Mini-Imagenet dataset.

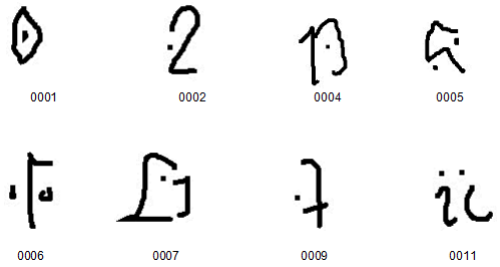

FIGURE 4. Omniglot: Partial images and labels Omniglot dataset.

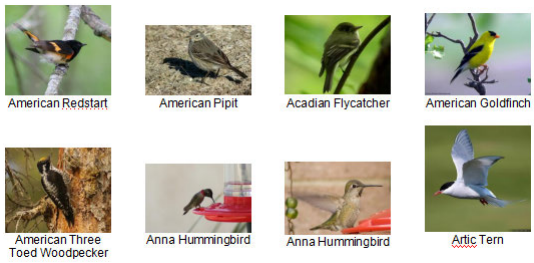

FIGURE 5. CUB: Partial images and labels CUB dataset.

comprise the training set, and the remaining 5,794 images are part of the test set.

B. IMPLEMENTATION DETAILS

The present implementation was based on the Python language and deep learning libraries of the Pytorch platform. We use two-branch Resnet18 encoders in our feature extraction module, which serves as a backbone in few-shot classification, prototype computing and feature extraction. To preprocess each image, we first resized it to 84 * 84, and then proceeded with centralization and standardization. The preprocessed image is fed into a two-branch encoder for feature extraction. The two-branch resnet18 decoder was then used, where one branch follows the model for parameter updating while the other freezes the parameters without following the model to learn. The two feature vectors were merged and then reduced to 640 by a fully connected layer and fused with the a priori statistics during the prediction stage.

C. EXPERIMENT SETTINGS

We trained and tested the proposed method on a PC platform equipped with an NIVIDA A100 GPU with 40 GB of memory and an Intel Xeon Gold 6248R CPU with 72GB with device. We utilized the traditional few-shot training setup of 5way-1shot and 5way-5shot respectively. During the training representation stage, we set the number of rounds as 100 and 150 for the test query stage. We uniformly employed 200 rounds and determined the experimental results by computing the average precision and 95 percent t-test confidence intervals. Regarding the artificial setting parameters, we chose the loss parameter margin of the comparison learning triad as 0.4, the weight of the total loss function λ as 0.3 and the number of nearest neighbour groups as 2 in the distribution correction stage.

It is worth mentioning that for the selection of prior information, we used directly migrated WideResNet [40] trained base class features in the overall process. However, beacause some information from Omniglot [38] was missing, we used the code method provided by Shuo Yang et al [12] to train on Omniglot [38] and use it as a prior feature. The experimental process comprises two phases. The first phase is the representation learning phase in which the double-branch ResNet18 is used as an encoder that combines comparative learning and prototype representation learning techniques to enable the encoder to learn the best representation. In the second phase, the model weights were fixed. The second stage is the test query stage, which uses the best model weights and fuses them with the distribution correction method for feature fusion to generate posteriori features. These features were then used as input to the logistic regression classifier provided by Scikit-learn for classification.

D. EVALUATION METRIC

In our proposed method,we used accuracy as the evaluation metric in our experiments, which is a widely used performance measure in few-shot classification tasks. To evaluate the performance of our proposed method in Mini-Imagenet [17], Omniglot [38], CUB [39] datasets, we randomly sample 100,000 episodes from test's datasets. While evaluating the classification accuracy of the model, we simultaneously calculated the 95 percent confidence interval for the mean accuracy. Based on these, we also selected the total parameters as tested methods as the performance measure of model complexity and efficiency. Traditional metrics for judging image classification problems include confusion matrix, ROC and AUC etc. in addition to accuracy. However, in the few-shot learning scenario, our support set and query set samples are generally in the situation of many sample categories but few samples and when calculating the ROC and AUC curves, the curves will jump too much and become unstable owing to the problem of too few samples. Therefore, when evaluating the model performance, we select the commonly used classification index called ACC.The ACC is shown below:

$$ACC = \frac{TP + TN}{TP + FN + FP + TN} \tag{12}$$

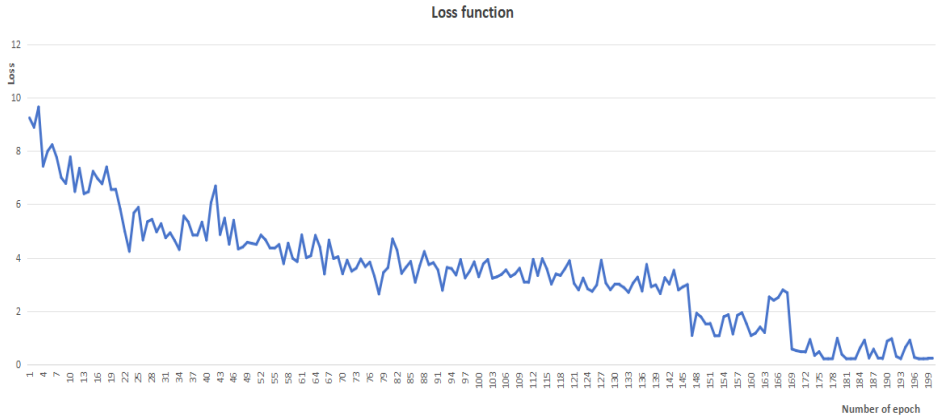

FIGURE 6. Loss function: Loss function change on Mini-Imagenet dataset when training epochs 200.

E. TEST RESULTS

For the comparison experiments, we used the publicly available code provided by the original authors of the paper to conduct the experiments in our experimental environment. Considering the differences in experimental environments, the results used in this paper are the numerical results obtained after conducting multiple experiments instead of directly adopting the values in the original paper provided by the papers. We make the experimental code publicly available at the following link: https://github.com/XuRanhui/Experiments-for-DCPNet/tree/main. This section discusses and analyzes the results of the comparative experiments in detail. Table 1 shows that our proposed method outperforms other classical methods in both datasets. This confirms its effectiveness. Despite this, it fails to reach the leading level on the Omniglot [38] dataset. However, the difference in accuracy does not exceed one percent in either case. We conjecture that Omniglot's [38] inaccuracy results in inappropriate information extraction during the WideResNet [40] feature extraction process. Secondly, the Omniglot [38] dataset is a handwriting dataset containing a large number of character corpora in different languages that have different shapes but similar structural features. While relation networks [9] mainly capture features by calculating the similarity of different samples, and thus perform better on the Omniglot [38] dataset, more complex datasets are not suitable for simple similarity calculation for metrics. Thus, it can be demonstrated that our DCPNet method is functional.

In contrast to other classical methods, our approach utilizes a two-branch structure for feature extraction that effectively blends the comprehensive features of shallow semantics and the specific features of deep semantics. By integrating the triplet loss and cross-entropy loss, the encoder, trained as per the dataset, can capture a prototypical representation that can pull different samples apart and similar samples together. When training the classifier, we utilized a distribution calibration method that facilitated fine-tuning in the fitting of classification boundaries by incorporating a priori features. To enhance the model performance for the few-shot classification problem, we made adjustments and improvements to both the prototype center and classification boundary.

TABLE 1. Results of comparative experiments on Mini-Imagenet.

Model	Mini-Imagenet	
	5way-1shot(%)	5way-5shot(%)
ProtoNet [18]	42.09 ± 0.62	65.69 ± 0.51
RelationNet [9]	49.12 ± 0.20	64.70 ± 0.55
MAML [8]	44.26 ± 0.42	59.90 ± 0.00
DeepEMD-Fcn [14]	64.66 ± 0.67	±
DeepEMD-Grid [14]	67.93 ± 0.67	±
DeepEMD-Sampling [14]	68.30 ± 0.29	±
DCPNet(ours)	**71.13 ± 0.63**	**86.44 ± 0.71**

TABLE 2. Results of comparative experiments on Omniglot.

Model	Omniglot	
	5way-1shot(%)	5way-5shot(%)
ProtoNet [18]	96.84 ± 0.26	99.27 ± 0.09
RelationNet [9]	**99.49 ± 0.07**	99.69 ±0.02
MAML [8]	92.72 ± 0.00	99.54 ± 0.87
DeepEMD-Fcn [14]	94.09 ± 0.26	±
DeepEMD-Grid [14]	95.83 ± 0.28	±
DeepEMD-Sampling [14]	95.68 ± 0.32	±
DCPNet(ours)	98.35 ± 0.25	**99.73 ± 0.03**

TABLE 3. Results of comparative experiments on CUB.

Model	CUB	
	5way-1shot(%)	5way-5shot(%)
ProtoNet [18]	48.32 ± 0.69	70.96 ± 0.58
RelationNet [9]	57.47 ± 0.33	71.29 ± 0.59
MAML [8]	47.56 ± 0.53	59.92 ± 0.61
DeepEMD-Fcn [14]	64.56 ± 0.32	±
DeepEMD-Grid [14]	72.73 ± 0.66	±
DeepEMD-Sampling [14]	75.32± 0.64	±
DCPNet(ours)	**76.82 ± 0.77**	**84.54 ± 0.37**

TABLE 4. Results of ablation experiments on Mini-Imagenet.

Method	5way-1shot(%)	5way-5shot(%)
Prototypical Network	42.09±0.62	65.69±0.51
DCPNet($without DC$)	57.35±0.62	73.24±0.52
DCPNet	71.13±0.63	86.44±0.71

F. ABLATION STUDY

Based on the comparison experiments, we conducted ablation experiments to assess the contribution of our approach featuring a two-branch encoder and a priori feature distribution calibration to the DCPNet. We first computed prototype centers for feature extraction using parallel hierarchical feature extraction module and few-shot differentiation loss to be used to train the prototype network, which we called DCPNet (without DC) and the results of the trained data were compared with the Prototypical Network and DCPNet. For our ablation experiments, we selected Mini-Imagenet [17], with a training setting of 5way-1shot and 5way-5shot.

The experimental ablation data indicates that the model accuracy is enhanced by approximately 15 percent using parallel hierarchical feature extraction module, few-shot differentiation loss and by about 14 percent from the a priori feature distribution calibration. This confirms the validity of our previously proposed approach.

V. CONCLUSION

In this paper, we present a parallel hierarchical feature extraction module, a few-shot differentiation loss function, and an improved distribution calibration for improving the representative features of prototype network metric learning and classifier boundaries. To address the limited feature extraction capabilities in prototypical network metric learning, we utilize a parallel hierarchical feature extraction module to integrate shallow and deep semantic information within images. This module is trained using the few-shot differentiation loss function, which fine-tunes metric learning to bring similar samples closer together while pushing dissimilar samples apart. In addition, we introduce an improved distribution calibration that incorporates statistical base class information with the original features. This calibration trains the classifier by sampling posterior features and ultimately refining the classifier boundaries. We evaluated the performance of our model on three publicly available datasets: Mini-Imagenet [17], Omniglot [38] and CUB [39]. The DCPNet model, which is our newly proposed approach, achieves highest results for two of these datasets. Notably, our method performs within one percent of the best method on the Omniglot dataset, highlighting its effectiveness. In terms of model architecture, our model utilizes a simple encoder design, resulting in a lightweight performance. Furthermore, the ablation experiments validate the contribution of each component to the overall performance of the model. According to our findings, each component plays a critical role in enhancing the performance of the model.

Despite the shortcomings observed in our experiments, this study offers an enhanced version of a prototypical network. Although the improvements made to the prototype representation and classification correction methods are significant, the approach still exhibits signs of algorithm redundancy and complexity. Furthermore, although the prototypical network achieved classification success in the distance space, it lacked a reasonable representation in the probability space. As such, our future work will focus on exploring methods to address the few-shot classification problem in the probability space, aiming to overcome these limitations.

COMPLIANCE WITH ETHICAL STANDARDS

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

REFERENCES

[1] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Generalizing from a few examples: A survey on few-shot learning," *ACM Comput. Surv.*, vol. 53, no. 3, pp. 1–34, May 2021.
[2] P. Sun, Y. Ouyang, W. Zhang, and X.-Y. Dai, "MEDA: Meta-learning with data augmentation for few-shot text classification," in *Proc. 13th Int. Joint Conf. Artif. Intell.*, Aug. 2021, pp. 3929–3935.
[3] Y. Jian and L. Torresani, "Label hallucination for few-shot classification," in *Proc. AAAI Conf. Artif. Intell.*, vol. 36, 2022, pp. 7005–7014.
[4] Y.-X. Wang, R. Girshick, M. Hebert, and B. Hariharan, "Low-shot learning from imaginary data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7278–7286.
[5] Y. Xian, S. Sharma, B. Schiele, and Z. Akata, "F-VAEGAN-d2: A feature generating framework for any-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10267–10276.
[6] C. Doersch, "Tutorial on variational autoencoders," 2016, *arXiv:1606.05908*.
[7] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: Introduction and outlook," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 4, pp. 588–598, 2017.
[8] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, Aug. 2017, pp. 1126–1135.
[9] Y. Zhuang, L. Tao, F. Yang, C. Ma, Z. Zhang, H. Jia, and X. Xie, "RelationNet: Learning deep-aligned representation for semantic image segmentation," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 1506–1511.
[10] W. Li, J. Xu, J. Huo, L. Wang, Y. Gao, and J. Luo, "Distribution consistency based covariance metric networks for few-shot learning," in *Proc. Conf. AAAI Artif. Intell.*, Jul. 2019, vol. 33, no. 1, pp. 8642–8649.
[11] A. Raghu, M. Raghu, S. Bengio, and O. Vinyals, "Rapid learning or feature reuse? Towards understanding the effectiveness of MAML," 2019, *arXiv:1909.09157*.
[12] S. Yang, L. Liu, and M. Xu, "Free lunch for few-shot learning: Distribution calibration," 2021, *arXiv:2101.06395*.
[13] X. Zhang, Y. Zhang, and Z. Zhang, "Multi-granularity recurrent attention graph neural network for few-shot learning," in *MultiMedia Modeling*, Prague, Czech Republic. Springer, 2021, pp. 147–158.
[14] C. Zhang, Y. Cai, G. Lin, and C. Shen, "DeepEMD: Differentiable Earth Mover's distance for few-shot learning," 2020, *arXiv:2003.06777*.
[15] R. Wang, H. Zheng, X. Duan, J. Liu, Y. Lu, T. Wang, S. Xu, and B. Zhang, "Few-shot learning with visual distribution calibration and cross-modal distribution alignment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 23445–23454.
[16] Z. Dai, J. Yi, L. Yan, Q. Xu, L. Hu, Q. Zhang, J. Li, and G. Wang, "PFEMed: Few-shot medical image classification using prior guided feature enhancement," *Pattern Recognit.*, vol. 134, Feb. 2023, Art. no. 109108.
[17] O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra, "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 1–9.

[18] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 30, 2017, pp. 1–11.
[19] T. Gao, X. Han, Z. Liu, and M. Sun, "Hybrid attention-based prototypical networks for noisy few-shot relation classification," in *Proc. 33rd Conf. Artif. Intell. (AAAI)*, vol. 33, 2019, pp. 6407–6414.
[20] S. Fort, "Gaussian prototypical networks for few-shot learning on omniglot," 2017, *arXiv:1708.02735*.
[21] B. Liu, Y. Cao, Y. Lin, Q. Li, Z. Zhang, M. Long, and H. Hu, "Negative margin matters: Understanding margin in few-shot classification," in *Computer Vision—ECCV*, Glasgow, U.K. Springer, 2020, pp. 438–455.
[22] V. N. Nguyen, S. Løkse, K. Wickstrøm, M. Kampffmeyer, D. Roverso, and R. Jenssen, "SEN: A novel feature normalization dissimilarity measure for prototypical few-shot learning networks," in *Computer Vision—ECCV*, Glasgow, U.K. Springer, 2020, pp. 118–134.
[23] J. Zhang, S. Li, X. Zhang, Z. Huang, and H. Miao, "Transductive semantic decoupling double variational inference for few-shot classification," *Image Vis. Comput.*, vol. 146, Jun. 2024, Art. no. 105034.
[24] S. X. Hu, D. Li, J. Stühmer, M. Kim, and T. M. Hospedales, "Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9058–9067.
[25] C. Ma, Z. Huang, M. Gao, and J. Xu, "Few-shot learning as cluster-induced Voronoi diagrams: A geometric approach," 2022, *arXiv:2202.02471*.
[26] A. Afrasiyabi, H. Larochelle, J.-F. Lalonde, and C. Gagné, "Matching feature sets for few-shot image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 9004–9014.
[27] M. Hiller, R. Ma, M. Harandi, and T. Drummond, "Rethinking generalization in few-shot classification," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 35, 2022, pp. 3582–3595.
[28] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, Dec. 2020.
[29] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "SimCLR: A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Learn. Represent.*, vol. 2, 2020, pp. 1–33.
[30] Y. Li, P. Hu, Z. Liu, D. Peng, J. T. Zhou, and X. Peng, "Contrastive clustering," in *Proc. AAAI Conf. Artif. Intell.*, May 2021, vol. 35, no. 10, pp. 8547–8555.
[31] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent-a new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 21271–21284.
[32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
[33] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9726–9735.
[34] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow twins: Self-supervised learning via redundancy reduction," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 12310–12320.
[35] B. Liang, Z. Chen, L. Gui, Y. He, M. Yang, and R. Xu, "Zero-shot stance detection via contrastive learning," in *Proc. ACM Web Conf.*, Apr. 2022, pp. 2738–2747.
[36] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.
[37] R. Debgupta, B. B. Chaudhuri, and B. K. Tripathy, "A wide resnet-based approach for age and gender estimation in face images," in *Proc. Int. Conf. Innov. Comput. Commun.* Springer, 2020, pp. 517–530.
[38] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, Dec. 2015.
[39] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD birds-200–2011 dataset," Tech. Rep., 2011.
[40] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2016, *arXiv:1605.07146*.

RANHUI XU was born in Dongying, Shandong, China, in 1999. He received the bachelor's degree in statistics from Jiaxing University, Jiaxing, China, in 2021. He is currently pursuing the master's degree in statistics with Shanghai University of Engineering Science, Shanghai, China. His research interests include data mining, machine learning, deep learning, and applications in computer vision.

KAIZHONG JIANG received the Ph.D. degree in systems analysis and integration from East China Normal University, Shanghai, China, in 2008. He is currently an Associate Professor with the School of Mathematics, Science and Statistics, Shanghai University of Engineering Science. He has published more than 30 journal articles. His main research interests include business web data mining, machine learning, information retrieval and knowledge retrieval, complex networks and complex systems, and algorithm research and design.

LULU QI was born in Dongying, Shandong, China, in 1998. She received the bachelor's degree in economic statistics from Qufu Normal University, Qufu, China, in 2020. She is currently pursuing the master's degree in statistics with Shanghai University of Engineering Science, Shanghai, China. Her research interests include machine learning and few-shot image classification and algorithm research and design.

SHAOJIE ZHAO received the bachelor's degree in economic statistics from Hainan University, Hainan, in 2021. He is currently pursuing the master's degree in statistics with Shanghai University of Engineering Science, Shanghai, China. His research interests include recommendation systems, natural language processing, and large language models.

MINGMING ZHENG was born in Mianyang, Sichuan, China, in 1998. He received the bachelor's degree in applied statistics from Chengdu College of Arts and Sciences, Chengdu, China, in 2021. He is currently pursuing the master's degree in statistics with Shanghai University of Engineering Science, Shanghai, China. His research interests include data mining, machine learning, deep learning, natural language processing, and algorithm research and design.