Artificial Intelligence Review
https://doi.org/10.1007/s10462-022-10223-3

Hierarchical few-shot learning based on coarse- and fine-grained relation network

Zhiping Wu[1,2] · **Hong Zhao**[1,2]

Abstract
Few-shot learning plays an important role in the field of machine learning. Many existing methods based on relation network achieve satisfactory results. However, these methods assume that classes are independent of each other and ignore their relationship. In this paper, we propose a hierarchical few-shot learning model based on coarse- and fine-grained relation network (HCRN), which constructs a hierarchical structure by mining the relationship among different classes. Firstly, we extract deep and shallow features from different layers at a convolutional neural network. The shallow feature information contains more common features among similar classes, while the deep feature information is more specific. The complementary of these different types of data features can effectively construct coarse- and fine-grained structures by clustering. Secondly, we design coarse- and fine-grained relation networks to classify according to the guidance of the hierarchical structure. The hierarchical class structure learned from data is important auxiliary information for classification. Experimental results show that HCRN can outperform several state-of-the-art models on the *Omniglot* and *miniImageNet* datasets. Especially, HCRN obtains 6.47% improvement over the next best under the 5-way 1-shot setting on the *miniImageNet* dataset.

Keywords Few-shot learning · Hierarchical structure · Coarse- and fine-grained · Relation network

1 Introduction

Few-shot classification (Koch et al. 2015; Miller et al. 2000) is a task where a classifier is expected to recognize new classes that are not seen in training by giving few samples (Feifei et al. 2006; Lake et al. 2011; Sung et al. 2018). For example, the image data of endangered species is difficult to obtain (Cheng et al. 2021). Establishing effective identification methods with few samples can bring great commercial and ecological value.

✉ Hong Zhao
hongzhaocn@163.com

[1] School of Computer Science, Minnan Normal University, Zhangzhou 363000, Fujian, China

[2] Key Laboratory of Data Science and Intelligence Application, Fujian Province University, Zhangzhou 363000, Fujian, China

Published online: 27 June 2022

Existing few-shot learning models are being well widely applied to various research fields such as computer vision (Alfassy et al. 2019; Liu et al. 2018; Tang et al. 2010), character recognition (Bertinetto et al. 2016; Fink 2005; Shyam et al. 2017), and image classification (Altae-Tran et al. 2017; Larochelle 2016; Triantafillou et al. 2017).

Distance calculation and relationship evaluation are often used to calculate the similarity between two samples in few-shot learning (Koch et al. 2015). In distance calculation, the closer the distance between samples, the higher the similarity (Xing et al. 2002). For example, Snell et al. (2017) utilized Euclidean distance to calculate similarity among query sample features and prototypes. Similarly, Vinyals et al. (2016) adopted Cosine distance to measure the distance among classes to determine whether they are similar. In particular, the quality of classes features affect the calculation of the distance among classes (Bateni et al. 2020; Zhao et al. 2021, 2019). For instance, Zhou et al. (2020) proposed a feature generation learning model, which makes features more distinguishable.

In relationship evaluation, the higher the sample relationship score is, the higher the similarity is Sung et al. (2018). For instance, Sung et al. (2018) proposed a relation network combining support and query set features to obtain a relationship score. Based on the relation network, Jiang and Wu (2021) adopted transfer learning technology to map the extracted features to an embedded common to measure the model. Clustering can gather the features of adjacent or similar classes to help classification (Chen et al. 2017; Ji et al. 2017; Yang et al. 2017). For example, Seo et al. (2020) embedded clustering space to narrow the gap among classes and obtain an effective class relationship. Similarly, Wen et al. (2021) integrated a transformer model into a prototypical network for more powerful relation-level feature extraction to obtain a higher relationship score.

The methods mentioned above make full use of fine-grained features and achieve effective results. However, these methods assume that fine-grained classes are independent of each other while ignoring their relationship. Figure 1(a) shows that the different fine-grained classes are uncorrelated and independent of each other. For instance, *sharks* and *seals* are independent of each other. There are generally exists correlations among classes. As shown in Fig. 1(b), there is a correlation among the same fine-grained classes that the distances among them are close to each other. For example, *whales* and *sharks* are similar in appearance, and

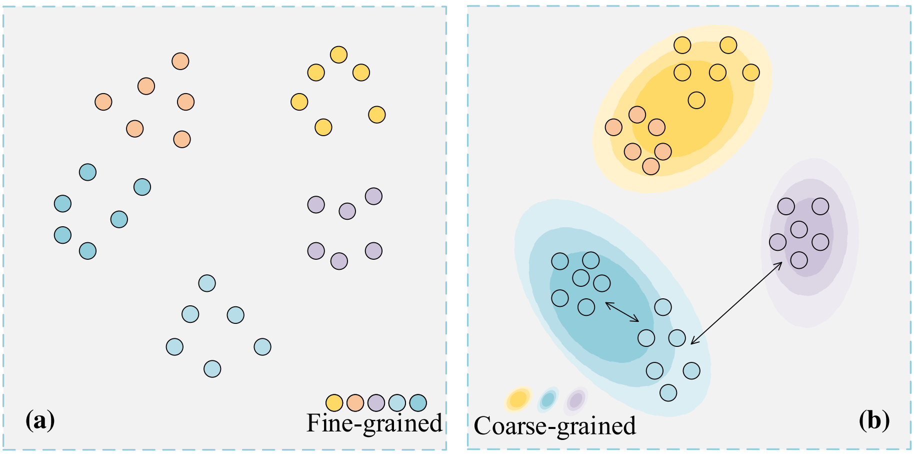


Fig. 1 A case to explain the basic idea of the HCRN. Different shapes represent different coarse- and fine-grained, while different colors represent different classes

the distance between their classes is closer than other classes. Fine-grained classes under the same coarse-grained class also correlate, with distances among them being smaller than those under other coarse-grained classes.

In this paper, we propose a hierarchical few-shot learning model based on relation network (HCRN), considering the relation among coarse- and fine-grained via clustering. We use a clustering method to make the same or similar classes closer and make the different or dissimilar classes farther away (Rahbar and Yazdani 2021; Zhang et al. 2021). First of all, we use Convolution Neural Networks (CNN) to extract shallow features of the image data in $\ell - i\,(1 \le i < \ell)$ layers and deep features in ℓ layer. Deep features are more specific and distinguishable, while shallow features contain similar features. We use the shallow features to construct coarse-grained classes via the spectral clustering method, and then the hierarchical structure among classes is obtained. This hierarchical structure is learned from data features rather than only considering the semantic structure of classes. Secondly, we obtain the features of different granularity according to the guidance of hierarchical structure knowledge. Then, we use coarse- and fine-grained relation networks to evaluate different granularities for distinguishing these classes.

The experiments are mainly carried out on the *Omniglot* and *miniImageNet* datasets. We compare HCRN with several methods to verify the feasibility. We use normalized mutual information and adjusted rand index to evaluate clustering measures comprehensively. In addition, we also use accuracy evaluation metrics. Experimental results show that HCRN, on the one hand, is comparable with several advanced models on the *Omniglot* dataset. In particular, HCRN achieves a satisfactory performance with 98.0% and 99.7% in 20-way 1-shot and 5-shot experimental settings. On the other hand, HCRN also achieves a satisfactory performance on the *miniImageNet* dataset. Especially in the 5-way 1-shot, HCRN obtains about 6.47% improvement over the next best. The major contributions of this paper are summarized as follows:

- We construct a feature extraction module, which can extract more discriminative deep and shallow features. Unlike the relational network that extracts a single feature for classification, this module makes full use of the complementarity of the deep and shallow features of the network.
- We propose a hierarchical classification model to handle the few-shot problem, which can process different coarse- and fine-grained within a single unified framework. Unlike traditional classification methods, which assume that classes are independent, the proposed model makes full use of the class hierarchy as auxiliary classification information.
- We evaluate our method on several few-shot learning datasets, finding it consistently achieves superior performance over existing approaches.

The remainder of this paper is organized as follows: Section 2 presents a brief survey on the related work of the HCRN. Section 3 describes the details of the proposed model. Section 4 introduces the experimental setup, including datasets, evaluation metrics, and comparison methods. The experimental results and analysis are given in Section 5. Finally, conclusions and future work are drawn in Section 6.

2 Related work

In this section, we briefly summarize related work into two categories: (1) granular computing and (2) coarse- and fine-grained clustering.

2.1 Granular computing

Granular computing is an underlying idea of using groups, classes, or clusters of elements called granules (Zadeh 1997). Granulation of the universe includes dividing the universe into subsets or grouping individual objects into clusters (Yao 2000). Granular computing is explicitly or implicitly used to solve various problems (Yao 2004), such as artificial intelligence, cluster analysis, and machine learning (Zadeh 1997). For example, Yao (2011) utilized a ternary theory to enrich the artificial intelligence view on the granular computing application. Similarly, Yao (2016) proposed a top-down progressive calculation, from the coarsest granularity to the smallest granularity. More specifically, the approximate solution in the coarser granulation process is fine-tuned in the next finer granulation process. In recent years, people are still very interested in granularity exploration (Qiu and Zhao 2022; Wang et al. 2022; Xu et al. 2022). A new measurement method is used to describe the uncertainty of classification to solve the problem of multi-granularity decision-making (Wang et al. 2022). According to the granularity idea, a large and difficult task can be divided into several small and controllable subtasks (Qiu and Zhao 2022).

2.2 Coarse- and fine-grained clustering

Clustering is an unsupervised technique that focuses on grouping data without knowing its labels in advance (Li et al. 2018a, b). The application of the clustering method in deep learning has made excellent progress in multi-view clustering (Zhang et al. 2017; Zheng et al. 2020) and coarse- and fine-grained (Do et al. xxxx; Fish et al. 2021; Wang et al. 2020; Zhang et al. 2021).

On the one hand, multi-view clustering is the clustering of views with the same label but different images to map new view representations (Zhan et al. 2017). For instance, Yu et al. (2021) used sparse subspace clustering to construct a similarity matrix and combined the similarity fusion strategy to fuse the fine-grained multiple views to improve the classification ability of the model. Another subspace clustering method clusters the granularity information of multiple views to obtain a new feature representation (Zheng et al. 2020). Similarly, there is a complementary relationship between the granularity of multiple views, where clustering can reconstruct the potential complementary relationship (Zhang et al. 2017).

On the other hand, the coarse- and fine-grained used in clustering can be grouped into three categories: (1) The affiliation between coarse- and fine-grained serves as auxiliary knowledge to guide the learning model (Zhang et al. 2021). For example, Wang et al. (2020) utilized the clustering method to locate the coarse-grained splicing position and then used the regional expansion or corrosion method to detect the fine-grained splicing position accurately. (2) The coarse- and fine-grained utilize the similarity between themselves to make the model perform better in classification (Fish et al. 2021). For instance, Do et al. (xxxx) effectively combined representation learning and clustering and used representation learning to extract the fine-grained features of image views. They maximized the mutual information among image views and then utilized the clustering method to form coarse-grained features. (3) The top-down hierarchical classification idea from coarse-grained to fine-grained is adopted to reduce the classification task (Cui et al. 2022). For example, Wang et al. (2021) proposed a progressive knowledge transfer method, which transfers coarse-grained knowledge constructed by clustering to fine-grained tasks.

Inspired by the above methods, we use the spectral clustering method to construct a tree structure and obtain coarse-grained to assist fine-grained classification according to the difference between the deep and shallow features of the network.

3 Hierarchical few-shot learning based on coarse- and fine-grained relation network

In this section, we introduce the framework and specific classification steps of HCRN in detail.

3.1 Basic framework

The basic flowchart of HCRN is shown in Fig. 2. This model is mainly composed of two parts:

(1) The first part is coarse- and fine-grained clustering based on shallow and deep features. The hierarchical clustering approach constructs a tree structure to help classify different granularities. The coarse-grained classes are clustered based on the similarities among the sample features of different fine-grained classes.

(2) The second part is hierarchical few-shot learning based on coarse- and fine-grained relation networks. The coarse-grained relation network is used to classify coarse-grained classes. The fine-grained network further classifies the fine-grained classes according to the classification of coarse-grained classes.

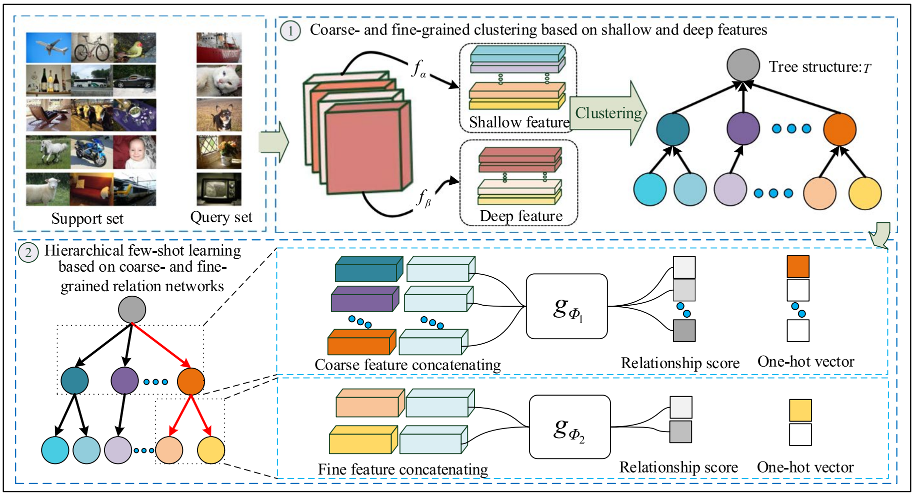


Fig. 2 Framework of the HCRN model

3.2 Coarse- and fine-grained clustering based on shallow and deep features

Feature extraction is an indispensable preprocessing step of classification (Asiri et al. 2021; He et al. 2020; Hui et al. 2019). Clustering different layers of features in CNN produce different clustering qualities and ultimately affects the classification effect (Bateni et al. 2020).

For the few-shot learning task, we simulate the few-shot learning setting through the episode-based training as proposed in Vinyals et al. (2016). Let x_i and x_j be the support and query set samples. In each iteration of training, randomly select C classes from training set, and use K samples of each class as the support set $S = \{x_i, y_i\}_{i=1}^{m}$, where $m = C \times K$ is the number of support set samples and y_i is the label of x_i. Similarly, we select K' samples of the remainder of those C classes serve as query set $Q = \{x_j, y_j\}_{j=1}^{n}$, where $n = C \times K'$ is the number of query set samples and y_j is the label of x_j. The support set has its own label space that does not intersect with the query set, as mentioned in Sung et al. (2018).

Let f_β and f_α be the deep and shallow features extraction modules, respectively. We fed samples into feature extraction module f_α and f_β, which obtain the shallow features $f_\alpha(x_i)$, $f_\alpha(x_j)$ and obtain the deep features $f_\beta(x_i)$, $f_\beta(x_j)$, where $x_i \in S$ and $x_j \in Q$. Parameters α and β are the network parameters of the shallow and deep feature extraction module, which are continuously optimized through the recursive loss during training.

We give an intuitive visualization of deep and shallow features from different layers of CNN in Fig. 3. We use GradCAM (Selvaraju et al. 2020) to show the attention range of samples at different layers of CNN. From this figure, we can obtain the following observations: (1) The features information extracted by classes is more specific in the deep layer of CNN. (2) The feature information extracted by class has similarities in the shallow layer of CNN. For example, the contour information extracted by the shallow features of CNN between *whale* and *shark*.

Then, we cluster the shallow features of the second output of the CNN full-connection layer. We use the spectral clustering method to construct coarse-grained classes

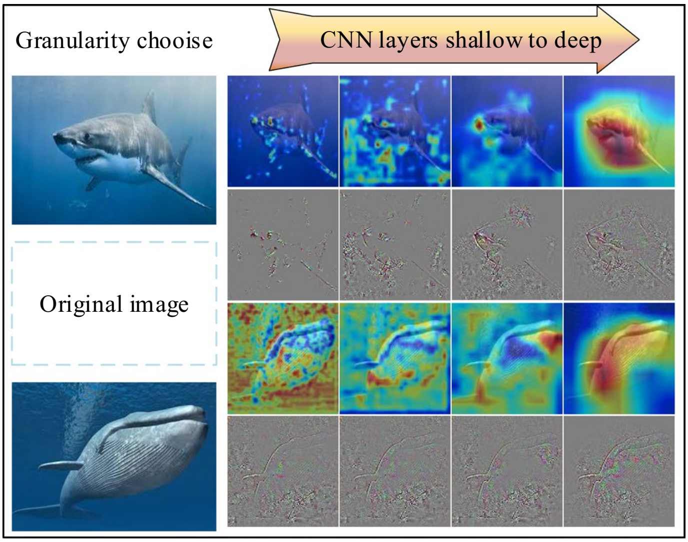


Fig. 3 Sample visualization by GradCAM in CNN different layers

according to the correlation of shallow features of similar classes. We build the tree structures by clustering shallow features of similar classes, as shown in Fig. 4.

The steps to build hierarchical structure knowledge are as follows: First, we extract the shallow features from CNN; Then, we construct class representative of the same class of samples by adding and averaging:

$$\mathbf{X}_c = \frac{1}{k}\sum_{i=1}^{k} f_\alpha(x_i), \tag{1}$$

where $\mathbf{X}_c$ is representative of the same class samples, c is the number of these classes representative, and k represents the number of samples with the same class. Similarly, we use deep features to build fine granularity:

$$\mathbf{Z}_h = \frac{1}{k}\sum_{i=1}^{k} f_\beta(x_i), \tag{2}$$

where $\mathbf{Z}_h$ represents fine-grained classes that are generated by adding and averaging deep features, h is the number of fine-grained classes, and k represents the number of the same class. Second, we adopt Euclidean distance to calculate the distance between any two classes and build a similarity matrix:

$$D_{ij} = \sqrt{\sum_{i,j\ (i\neq j)}^{c} (\mathbf{X}_i - \mathbf{X}_j)^2}, \tag{3}$$

where c is the number of classes and D_{ij} represents calculation of Euclidean distance between $\mathbf{X}_i$ and $\mathbf{X}_j$. Then, we construct the adjacency matrix and degree matrix according to the similarity matrix to calculate the Laplacian matrix. Finally, we use k-means to calculate the degree of correlation between classes. We take advantage of spectral clustering method to obtain coarse-grained classes $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \cdots, \mathbf{X}^u\}$ that represent similar or similar fine classes. We set the number of clusters u to achieve the spectral clustering method's best effect. Similarly, the spectral clustering method can automatically generate clusters, but it is uncontrollable and easy to mislead the classification. It means that the number of clusters obtained by setting the spectral clustering method can better represent the aggregation of similar fine-grained classes.

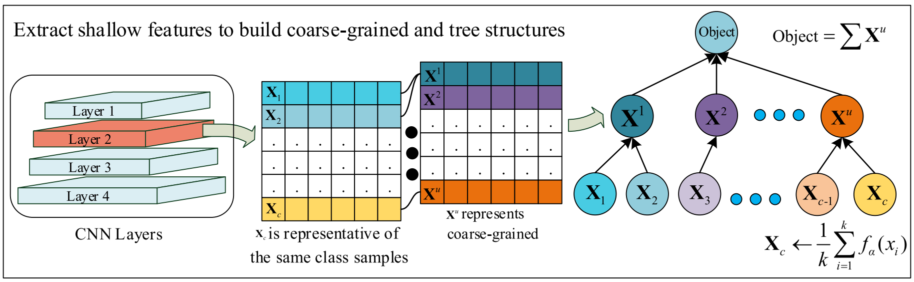


Fig. 4 Coarse-grained clustering

3.3 Hierarchical few-shot learning

We can divide the classification task into subtasks by tree structure. We combine knowledge embodied in the hierarchical tree constructed and extract different granular features of the support and query set from different layers of CNN. We concatenate coarse-grained features of the query set with coarse-grained classes obtained by clustering to obtain their relationship score:

$$r_{c_{ij}} = g_{\phi_1}(C(\mathbf{X}^i, f_\alpha(x_j))), \tag{4}$$

where $r_{c_{ij}}$ is the score of the relationship between coarse-grained features and shallow features of query set, $\mathbf{X}^i$ is represent the i-th coarse-grained classes, $i = \{1, \cdots, u\}$, $x_j \in Q$, $j = \{1, \cdots, n\}$, f_α is shallow feature extraction network, $C(\cdot, \cdot)$ is splicing between two features, and g_{ϕ_1} is a coarse-grained relation network. So that the g_{ϕ_1} network can learn better parameters, we use the $L(\cdot, \cdot, \cdot)$ function to predict coarse-grained classes label by relationship score and tree structure T construct:

$$y_{c_j} \leftarrow L(r_{c_{ij}}, x_j, T). \tag{5}$$

We can obtain the coarse-grained classes of the query set according to $r_{c_{ij}}$. We can know which fine classes $\mathbf{Z}_v$ $(v = 1, \cdots, h)$ are under the coarse-grained classes via the knowledge embodied in the hierarchical structure. Then, we concatenate fine-grained features of the query set with fine-grained features in the fine-grained class to obtain their relationship scores:

$$r_{f_{ij}} = g_{\phi_2}(C(\mathbf{Z}_i, f_\beta(x_j))), \tag{6}$$

where $r_{f_{ij}}$ is the score of the relationship between fine-grained features and deep features of query set, $i = \{1, \cdots, v\}$, $\mathbf{Z}_v$ is the fine-grained classes under coarse-grained classes predicted by $r_{c_{ij}}$, $\mathbf{Z}_v$ is a subset of $\mathbf{Z}_h$, $v = \{1, \cdots, h\}$, $x_j \in Q$, $j = \{1, \cdots, n\}$, f_β is deep feature extraction network, $C(\cdot, \cdot)$ is splicing between two features, and g_{ϕ_2} is a fine-grained relation network. Similarly, according to the relationship score $r_{f_{ij}}$ and the tree structure T, we use the L function to predict the fine label:

$$y_{f_j} \leftarrow L(r_{f_{ij}}, x_j, T). \tag{7}$$

We use mean square error loss (Eq. (8)) to train our coarse-grained relation model, regressing the relationship score $r_{c_{ij}}$ to the ground truth (matched pairs have similarity 1 and the mismatched pairs have similarity 0):

$$\underset{\alpha, \phi_1}{argmin}\ \mathcal{L}_c = (\sum_{i=1}^{u} \sum_{j=1}^{n} (r_{c_{ij}} - \mathbf{1}(y_{c_i} == y_{c_j}))^2), \tag{8}$$

where $\mathcal{L}_c$ is the coarse-grained classification loss, y_{c_i} is coarse-grained class label, and y_{c_j} is coarse-grained class prediction label of query set. Similarly, we use the mean square error loss (Eq. (9)) to train our fine-grained relation model and minimize the parameters β and ϕ_2 to constantly approach the final optimization objective function:

$$\underset{\beta,\phi_2}{argmin}\ \mathcal{L}_f = \left(\sum_{i=1}^{v}\sum_{j=1}^{n}(r_{f_{ij}} - \mathbf{1}(y_{f_i} == y_{f_j}))^2\right), \tag{9}$$

where $\mathcal{L}_f$ is the fine-grained classification loss, y_{f_i} is the fine-grained class label, and y_{f_j} is the fine-grained class prediction label of the query set. The model needs to identify the coarse-grained query samples before identifying the fine-grained query samples. The proportion of coarse-grained loss $\mathcal{L}_c$ and fine-grained $\mathcal{L}_f$ loss affects the final classification performance of the model. Finally, we use the loss function $\mathcal{L}$ to control parameters of the two in a more reasonable range and converge our loss function:

$$\mathcal{L} = \min_{\alpha,\beta,\phi_1,\phi_2}(\lambda\mathcal{L}_c + (1-\lambda)\mathcal{L}_f). \tag{10}$$

A hierarchical few-shot learning based on coarse- and fine-grained relation network detailed procedure is listed in Algorithm 1, which provides the training model and pseudocode for a training episode.

Algorithm 1 Hierarchical few-shot learning based on coarse- and fine-grained relation network (HCRN)

Input: Training set $\mathcal{D} = \{(x_1, y_1), ..., (x_M, y_M)\}$ ($y \in [1, 2, ..., N]$), where M, N mean the number of samples and classes in the training set.
Output: The loss $\mathcal{L}$ for a randomly generated training episode.

1: Select randomly C classes and k samples of each class to construct support set $S = \{(x_1, y_1), ..., (x_m, y_m)\}$ and query set $Q = \{(x_1, y_1), ..., (x_n, y_n)\}$ separately;
2: Initialize network f_α, f_β, g_{ϕ_1}, g_{ϕ_2};
3: **for** $i = 1 : m$ **do**
4: $\quad X_c \longleftarrow \frac{1}{k}\sum_{i=1}^{k} f_\alpha(x_i)$ by shallow feature;
5: $\quad$ Obtain the fine-grained $Z_h \longleftarrow \frac{1}{k}\sum_{i=1}^{k} f_\beta(x_i)$ by deep feature;
6: $\quad$ Obtain the coarse-grained $\mathbf{X}$ by spectral clustering method;
7: $\quad$ **for** $j = 1 : n$ **do**
8: $\quad\quad$ Compute coarse-grained score $r_{c_{ij}}$ according to Eq. (4);
9: $\quad\quad$ Obtain fine-grained and number of fine-grained (v);
10: $\quad\quad$ Update parameter ϕ_1 according to Eq. (8);
11: $\quad\quad$ **while** $v > 0$ **do**
12: $\quad\quad\quad$ Compute fine-grained score $r_{f_{ij}}$ according to Eq. (6);
13: $\quad\quad\quad$ Update parameter ϕ_2 according to Eq. (9);
14: $\quad\quad$ **end while**
15: $\quad\quad$ Update parameters α and β according to Eq. (10);
16: $\quad\quad$ Return backward loss $\mathcal{L}$;
17: $\quad$ **end for**
18: **end for**

In particular, we construct the coarse- and fine-grained listed in lines 5 and 6. The coarse-grained relationship score computation is listed in line 8. According to the coarse-grained relationship score, we further explore the fine-grained to which the sample belongs. Therefore, we compute the fine-grained relationship score listed in line 12, and update the classification loss listed in line 16. Two popular few-shot learning models inspired our model which are Relation network (Sung et al. 2018) and Prototype network

(Snell et al. 2017). Therefore, similar network structures and experimental settings were adopted in HCRN. Moreover, we were inspired by the calculation of the final confidence score of related concepts in Li et al. (2019). Our classification is completed by calculating the coarse- and fine-grained relationship scores, which simplify the model to a certain extent. In addition, our model is at the same level as the few-shot learning models based on the Relation network in terms of time complexity.

4 Experimental settings

In this section, firstly, we describe *Omniglot* and *miniImageNet* datasets used in the experiment. Secondly, we introduce evaluation metrics and comparison methods used in the experiment. To prove the effectiveness of our method, we set the learning rate, training episode, training batch, and most of the parameters the same to benchmark level (Sung et al. 2018). For example, the optimizer is Kingma and Ba (2015) with an initial learning rate of 10^{-3} and reduced the learning rate by half for every 10, 000 episodes. We extract shallow features from the second layer of CNN and deep features from the fourth layer of CNN. The parameter λ controls the proportion of coarse- and fine-grained participation in the classification process. We set parameter $\lambda = 0.65$ in the 5-shot experiments on the *Omniglot* and *miniImageNet* datasets. We set the number of clusters from the set $\{2, 3,..., 12\}$ in the 20-way 5-shot experiments on the *miniImageNet* dataset. In test settings, we conduct a few-shot classification on 2,000 random episodes samples from the test set and compute the mean accuracy together. Our model uses end-to-end training from scratch. All experiments are performed on a Ubuntu20.04 desktop computer using NVIDIA GTX3090 with 24.0 GB video memory and a 2.40GHz $\times$ 24 Intel Xeon Silver 4214R CPU. This study's basic data and code have been uploaded to GitHub and can be accessed via the following link: https://github.com/fhqxa/HCRN.

4.1 Datasets

In our experiments, we use two datasets, including *Omniglot* and *miniImageNet*. The details are as follows:

Omniglot (Lake et al. 2011) includes 1623 characters (classes) from 50 different alphabets. Each class includes 20 samples drawn by different people. We use 423 original classes plus rotations for test and the remaining 1200 classes plus rotations for training. All input images are set to 28×28. Our experimental setup for training is based on reference, in each episode (Sung et al. 2018), in each training episode, each class with the 20-way 1-shot has 10 query images, and the 20-way 5-shot has 5 query images. At the same time, we also set the 5-way 5-shot to have 15 query images, and the 5-way 1-shot contains 19 query images. In one training episode/mini-batch for the 20-way 1-shot experiments, there are $10 \times 20 + 1 \times 20 = 220$ images.

miniImageNet (Vinyals et al. 2016) consists of 60,000 color images with 100 classes, and each class contains 600 samples. Referring to the split proposed in Larochelle (2016), we split 50 classes for training and 50 classes for the test, respectively. All input images are set to 84×84. Our training follows the standard setting utilized by most existing few-shot learning work. We set 5-way 1-shot and 5-way 5-shot experiments, which is the standard setting utilized by the few-shot learning work proposed in Sung et al. (2018). In each training episode, each of the C sample classes with the 5-way 5-shot has 10 query images, and the 5-way 1-shot contains 15 query images in each training episode. There are $20 \times 5 + 20 \times 5 = 200$ images in one training episode/mini-batch for the 20-way 5-shot experiments.

4.2 Evaluation metrics

In our experiments, we use three evaluation metrics: Accuracy (ACC), Adjusted Rand Index (ARI), and Normalized Mutual Information (NMI). We use NMI and ARI to evaluate the clustering effect of extracting different granularity features at different network layers. Specifically, the range of these measures is [0,1], and higher scores indicate more accurate clustering results. To evaluate the clustering effect of the proposed model on a small sample and compare the objective clustering performance of the model, we are adopting a widely used metric NMI which is defined as follows:

$$NMI(y_i, y_j) = \frac{I(y_i, y_j)}{\sqrt{H(y_i)H(y_j)}}, \tag{11}$$

where $y_i = \{y_{c_i}, y_{f_i}\}$ is the ground truth label, $y_j = \{y_{c_j}, y_{f_j}\}$ is the cluster label, $H(:)$ stands for the entropy, and $I(y_i, y_j) = H(y_i) - H(y_i \mid y_j)$ denotes the mutual information. For the hierarchical few-shot learning, we define the accuracy as follows:

$$ACC = \max \frac{\Sigma_{i=1}^{n} \mathbf{1}\{y_i == y_j\}}{n}, \tag{12}$$

where $\mathbf{1}\{y_i == y_j\} = 1$ when $y_i = y_j$, otherwise $\mathbf{1}\{y_i == y_j\} = 0$.

4.3 Comparison methods

In this subsection, we compare HCRN with several few-shot learning methods. The details of them are introduced as follows:

(1) Matching Network (Vinyals et al. 2016): This method proposes metric learning based on deep neural features and augments neural networks with external memories.
(2) Prototypical Network (Snell et al. 2017): This method automatically generates the prototypes of each class in the support set and then leverages Euclidean distance to measure the distance among these prototypes and query samples.
(3) MAML (Finn et al. 2017): This approach aims to train the initial parameters of the model so that the model would perform new tasks well after only one or several steps of gradient descent update.
(4) Relation Network (Sung et al. 2018): This approach proposes a relation network that defines the relationship among query and support samples as a relation score.
(5) AttWeightGen (Gidaris and Komodakis 2018): This method redistributes class weights based on the attention mechanism, which not only improves the classification accuracy but also has a good effect on the feature representation.
(6) TADAM (Oreshkin et al. 2018): This approach changes the essence of the few-shot learning parameter update by designing an adaptive metric between classes and improves the performance of the few-shot learning algorithm.
(7) LEO (Rusu et al. 2019): This method embeds the optimization model to solve the problem of difficult decoupling of model parameters in high-dimensional space.

(8) E3BM (Liu et al. 2020): This approach proposes meta-learn the ensemble of epoch-wise empirical Bayes models to achieve robust predictions.

5 Experimental results and analysis

In this section, we present the experimental results and discussions from four perspectives: (1) we compare different granular layer's performance; (2) we analyze different clustering number's performance; (3) we compare the performance of different parameter λ values; and (4) we compare HCRN with several models.

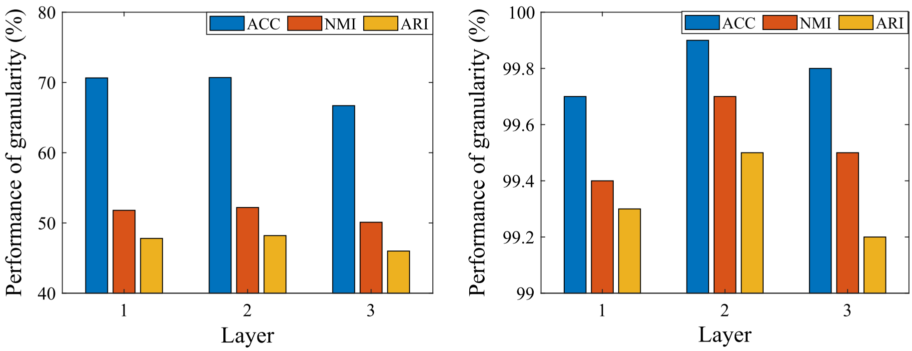
5.1 Performance comparison of different granular layers

In this section, we exploit the influence of the output features of different CNN layers on the effectiveness of HCRN. The following comparative experiments under the *miniImageNet* and *Omniglot* datasets are conducted in 5-way 1-shot. We set parameters λ=0.65 and K=5. The performance of the *miniImageNet* and *Omniglot* datasets in different layers of CNN are shown in Fig. 5.

Fig. 5(a) shows the performance of the *miniImageNet* dataset in different layers of CNN, and we can observe the followings:

(1) The ACC of the first and second layers are 70.65% and 70.70%, respectively, while the third layer is 66.70%. The NMI of the second layer is 52.25%, which is 0.47% and 2.13% higher than that of the first layer and the third layer, respectively. The features of CNN's shallow output are similar, but the clustering results of different layers are different. That's why the ACC of the first layer and the second layer are almost the same, but the NMI is different.

(2) Similarly, the ARI of the second layer is 0.44% and 2.21% higher than that of the first layer and the third layer, respectively. The experimental results show that the features of the second layer of CNN can reflect the coarse-grained information when using *miniImageNet* dataset.



(a) Performance of the *miniImageNet* dataset **(b)** Performance of the *Omniglot* dataset

Fig. 5 Performance comparison with different layers of CNN: (a) Performance of the *miniImageNet* dataset on different layers of CNN; (b) Performance of the *Omniglot* dataset on different layers of CNN

Fig. 5(b) shows the performance of the *Omniglot* dataset in different layers of CNN, and we can observe the following:

(1) The ACC of the second layer is 99.9%, which is 0.2% and 0.1% higher than that of the first and third layers, respectively. The NMI and ARI in the second layer are higher than those in the first and third layers. The experimental results show that the features of the second layer of CNN can represent the coarse-grained information when using the *Omniglot* dataset.
(2) In addition, the performance of the *Omniglot* dataset in different layers of CNN is better than the *miniImageNet* dataset. The results demonstrate that the second layer features of CNN are more suitable for clustering to obtain coarse-grained classes.

5.2 Performance comparison of different clustering numbers

In this section, we describe the influence of the number of clusters on classification accuracy. Firstly, we use a toy dataset to give an intuitive interpretation of clustering our clustering model. As shown in Fig. 6, the toy dataset consists of four classes: *flat fish*, *shark*, *whale*, and *seal*. Among them, *flat fish* and *shark* are *fish*, while *whale* and *seal* are *aquatic mammals*. We find that the *shark* and the *whale* are clustered together through the clustering model. We use the clustering method to break the semantic gap. Our model can bring together classes with the same or similar appearance.

Then, we further explore the impact of the number of clusters on the classification effect. We randomly select 20 of the 50 classes test set to perform the 5-shot experiment on the *miniImageNet* dataset. We set the parameter λ= 0.65 and set the parameter k from the set {2, 3,..., 12}. Table 1 shows the performance of 20-way clustering on the *miniImageNet* dataset based on different parameter k. The best results are shown in highlighted.

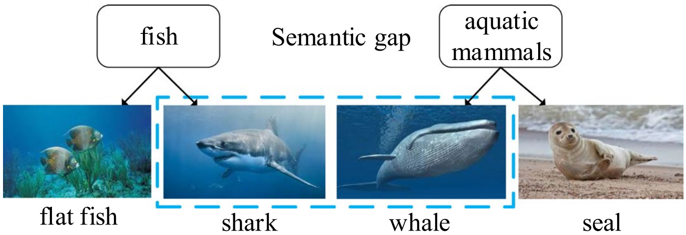


Fig. 6 An example diagram describing the semantic gap

Table 1 The performance of 20-way clustering on the *miniImageNet* dataset based on different parameter k (%)

metrics	k										
	2	3	4	5	6	7	8	9	10	11	12
ACC	31.00	29.03	26.14	27.22	**31.06**	26.71	27.00	25.10	29.00	27.99	25.03
NMI	52.42	55.44	45.69	49.20	**57.00**	51.17	48.17	46.56	50.24	56.78	44.43

The performance of clustering is the best when k=6. The ACC and the NMI are 31.06% and 57.00%, respectively, which are 0.06% and 0.22% higher than the suboptimal results. The experimental results show that manually setting the number of clusters for spectral clustering can help to improve the quality of clusters and improve classification accuracy. This also means that clusters of similar fine-grained classes can be better represented by setting the number of clusters.

Secondly, we use Example 1 to explain the hierarchical tree structure intuitively.

Example 1 We visualize the clustering results shown in Fig. 7. Our clustering model brings together classes with similar appearances. Through the tree structure, we can reduce the calculation of classification tasks. For example, it needs comparisons $6 + 4 = 10$ times when the type of test sample is *mushroom*. Especially, it needs to be carried out 20 times, in the case without the tree structure (flat situation), because the test sample needs to be compared with each class.

Finally, we compare the hierarchical performance of HCRN with the relation network. The performance of 20-way 5-shot clustering on the *miniImageNet* dataset is shown in Table 2. Compared with Relation Network, HCRN classification accuracy gains about 2.05% improvements. The experimental results show that the coarse- and fine-grained produced by clustering is helpful for classification. Throughout the above four-part analysis, our hierarchical clustering strategy is feasible.

5.3 Performance comparison of different parameter λ values

In this section, we discuss the effectiveness of coarse-grained and fine-grained participation and explore the influence of parameter λ. We evaluate whether coarse-grained

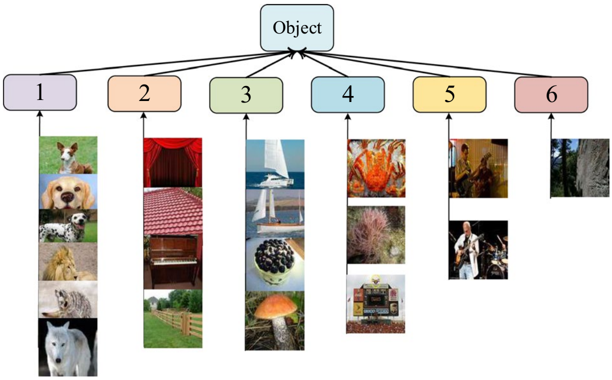


Fig. 7 An example of constructing tree structure (20-way 5-shot, k=6)

Table 2 The performance of 20-way 5-shot clustering on the *miniImageNet* dataset (%)

Model	Hierarchical	20-way 5-shot
Relation Network (Sung et al. 2018)	N	29.01
HCRN	Y	31.06

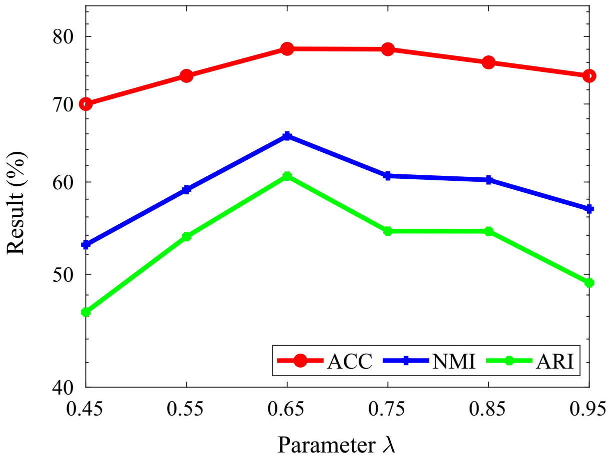


Fig. 8 Performance of parameter λ on the *miniImageNet* dataset

participation contributes to the final classification of our model. We utilize parameter λ to control the proportion of coarse- and fine-grained features in the model. We use 5-way 5-shot setting to experiment on the *miniImageNet* dataset. We set the parameters k=5 and λ from the set {0.45, 0.55,..., 0.95}. Except for λ values, the settings of all the experiments keep the same.

Fig. 8 shows the effect of different parameters λ on the classification results on the *miniImageNet* dataset, and we can obtain the following observations:

(1) The NMI increases from 53.01% to 65.72% when the parameter λ increases from 0.45 to 0.65. Similarly, the results of ACC and ARI improve, respectively. With the increasing proportion of coarse-grained in the model, the classification effect of the model is also improving.

(2) The ACC decreases from 78.06% to 74.00% when the parameter λ increases from 0.65 to 0.95. Similarly, the results of NMI and ARI decrease, respectively. Coarse-grained is too prominent in the model, ignoring the role of fine-grained, which reduces the classification and clustering effect of the model.

Figure 9 shows the effect of parameters $\lambda = 0$ and $\lambda = 0.65$ on the classification results on the *miniImageNet* dataset, and we can obtain the following observation:

(1) The ACC is 65.99% when $\lambda = 0$ while the ACC is 78.06% when $\lambda = 0.65$. Similarly, when $\lambda = 0.65$, NMI and ARI are higher than those when $\lambda = 0$, respectively.

(2) The coarse-grained participation in the model affects the classification and clustering effects. The experimental results show that the model's proportion of coarse- and fine-grained reaches an appropriate state when λ=0.65.

5.4 Comparison with different models

In this section, we compare HCRN with several state-of-the-arts few-shot classification methods on the *Omniglot* and *miniImageNet* datasets. Following (Sung et al. 2018), we use the same experiment setting for the 1-shot and 5-shot experiments. We set the parameters $\lambda = 0.65$ and $k = 6$ in 20-way experiment. Note that the experimental results from

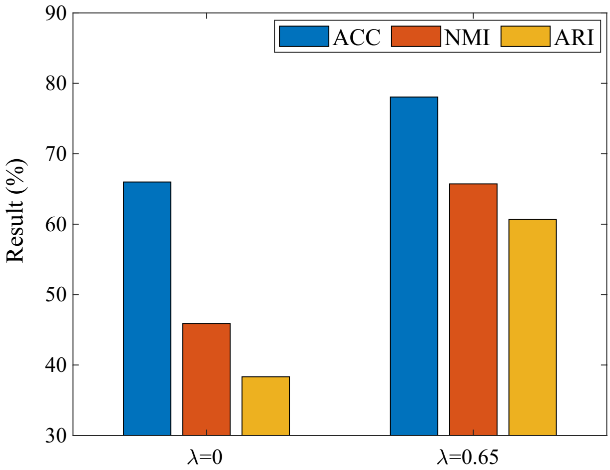


Fig. 9 Performance of parameter λ participate

the original author are represented by "$*$". We reproduced Relation Network (Sung et al. 2018) in the experimental environment of our machine. The best-performing method is highlighted, and "-" indicates that the method is not reported.

The experimental results of the proposed HCRN and state-of-the-art models are shown in Table 3, and we can obtain the following observations:

(1) Under the 5-way 1-shot and 5-shot experiments, HCRN achieves 99.9% and 99.9%, which are 0.1% higher than Relation Network in 1-shot. HCRN achieves 98.0% and 99.7%, which are 0.2% and 0.5% higher than Relation Network in 5-shot, under 20-way 1-shot, and 5-shot experiments. Thus, compared with other models, HCRN has a good ability for classification discrimination in the 5/20-way 1-shot and 5-shot experiments of the *Omniglot* dataset.

Table 3 Few-shot classification accuracy on the *Omniglot* dataset (%)

Method	Fine	5-way		20-way	
		1-shot	5-shot	1-shot	5-shot
Mann* Santoro et al. (2016)	N	82.8	94.9	–	–
Convolutional Siamese Nets* Koch et al. (2015)	N	96.7	98.4	88.0	96.5
Convolutional Siamese Nets* Koch et al. (2015)	Y	97.3	98.4	88.1	97.0
Matching Network* Vinyals et al. (2016)	N	98.1	98.9	93.8	98.5
Matching Network* Vinyals et al. (2016)	Y	97.9	98.7	93.5	98.7
Siamese Nets with Memory* Kaiser et al. (2017)	N	98.4	99.6	95.0	98.6
Neural Statistician* Edwards and Storkey (2017)	N	98.1	99.5	93.2	98.1
Meta Nets* Munkhdalai and Yu (2017)	N	99.0	–	97.0	–
Prototypical Network* Snell et al. (2017)	N	98.8	99.7	96.0	98.9
MAML* Finn et al. (2017)	Y	99.1	**99.9**	96.1	99.1
FBM* Yang et al. (2022)	N	**99.9**	**99.9**	96.8	99.1
Relation Network Sung et al. (2018)	N	99.8	**99.9**	97.8	99.2
HCRN	N	**99.9**	**99.9**	**98.0**	**99.7**

(2) We achieve state-of-the-art performance under all experiments with higher accuracy, except for the 5-way 5-shot, where our model accuracy is consistent with MAML, FBM, and Relation Network. MAML has more complex mechanisms or fine-tunes on the target problem, while we did not do that. HCRN is better than FBM in 20-way 1shot and 5-shot experiments. The experimental results show the feasibility of the HCRN model on the *Omniglot* dataset.

We compute few-shot classification accuracy on the *miniImageNet* dataset. All the comparative experiments and our models are based on CNN. We set the parameters $\lambda = 0.65$ and $k = 5$ on 5-way 5-shot experiment.

The performance of the proposed HCRN and state-of-the-art models in the 5-way 5-shot accuracy on the *miniImageNet* dataset are shown in Table 4, and we can obtain the following observations:

(1) HCRN achieves 70.67% classification accuracy for 5-way 1-shot experiments on the *miniImageNet* dataset. HCRN is 6.57% outperforms the best model and 8.83% higher than the next best model, confirming that HCRN is effective.
(2) Compared with LEO and LDGP, HCRN has 8.8% and 14.07% improvement on the 5-way 1-shot experiment. Compared with LDGP, HCRN has a 5.16% improvement in the 5-way 5-shot experiment. HCRN is 2.48% lower than E3BM, mainly caused by two factors. On the one hand, E3BM uses hyperparameters generated by hyper prior learners conditional on task-specific data, while our model uses the end-to-end learning method. On the other hand, it also shows the effectiveness of our model in few-shot learning.

6 Conclusions and future work

This paper proposed a hierarchical clustering with coarse- and fine-grained model based on relation network for few-shot learning (HCRN). We utilized the complementary characteristics between the deep and shallow features to construct the coarse- and fine-grained. We made full

Table 4 Few-shot classification accuracy on *miniImageNet* (%)

Method	Fine	5-way	
		1-shot	5-shot
Matching Network* Vinyals et al. (2016)	N	44.40	56.04
Meta Nets* Munkhdalai and Yu (2017)	N	50.17	–
Meta-learn LSTM* Larochelle (2016)	N	44.21	61.31
Prototypical Network* Snell et al. (2017)	N	50.54	64.03
MAML* Finn et al. (2017)	Y	50.20	68.86
Relation Network Sung et al. (2018)	N	51.26	66.02
AttWeightGen* Gidaris and Komodakis (2018)	N	57.06	73.43
TADAM* Oreshkin et al. (2018)	N	58.80	77.00
LEO* Rusu et al. (2019)	N	61.84	77.71
E3BM* Liu et al. (2020)	N	64.20	**80.54**
LDGP* Wang et al. (2021)	N	56.60	72.90
HCRN	N	**70.67**	78.06

use of the similarity relation of classes and the subordination relation among coarse- and fine-grained classes in the hierarchical structure. Our hierarchical structure considers the relationship among similar granularities and the relationship between different layers of granularity. In addition, experiments show that HCRN can effectively solve the semantic gap in the semantic hierarchy. The experimental results demonstrate that HCRN achieves more effective performance than other few-shot learning models. In future work, we will further build an adaptive model for different parameters to improve the generalization ability of the model. Furthermore, we will further explore the hierarchical model on a larger dataset.

Acknowledgements This work was supported by the National Natural Science Foundation of China under Grant No. 62141602 and the Natural Science Foundation of Fujian Province under Grant No. 2021J011003.

Author Contributions Methodology, software, data curation, and writing-original draft were done by Zhiping Wu. Validation, methodology, reviewing, and editing were done by Hong Zhao.

Funding Existing funding was provided in acknowledgment.

Compliance with ethical standards

Conflict of interest All authors declare that there is no conflict of interest in this manuscript.

Ethical approval This paper does not contain any studies with human participants performed by any of the authors. This material is the authors' original work, which has not been previously published elsewhere. The paper is not currently being considered for publication elsewhere. The paper reflects the authors' research and analysis in a truthful and complete manner.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

Alfassy A, Karlinsky L, Aides A, Shtok J, Harary S, Feris R, Giryes R, Bronstein AM (2019) Laso: label-set operations networks for multi-label few-shot learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 6548–6557

Altae-Tran H, Ramsundar B, Pappu AS, Pande V (2017) Low data drug discovery with one-shot learning. ACS Cent Sci 3(4):283–293

Asiri N, Bchir O, Ismail MMB, Zakariah M, Alotaibi YA (2021) Image-based smoke detection using feature mapping and discrimination. Soft Comput 25(5):3665–3674

Bateni P, Barber J, Meent JWVD, Wood F (2020) Improving few-shot visual classification with unlabelled examples. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 14481–14490

Bateni P, Goyal R, Masrani V, Wood F, Sigal L (2020) Improved few-shot visual classification. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 14493–14502

Bertinetto L, Henriques JF, Valmadre J, Torr PH, Vedaldi A (2016) Learning feed-forward one-shot learners. In: International Conference on Neural Information Processing Systems, pp 523–531

Chen D, Lv J, Yi Z (2017) Unsupervised multi-manifold clustering by learning deep representation. In: AAAI Conference on Artificial Intelligence

Cheng C, Li C, Han Y, Zhu Y (2021) A semi-supervised deep learning image caption model based on pseudo label and n-gram. Int J Approx Reason 131:93–107

Cui Y, Liao Q, Hu D, An W, Liu L (2022) Coarse-to-fine pseudo supervision guided meta-task optimization for few-shot object classification. Pattern Recogn 122:108296

Do K, Tran T, Venkatesh S (2021) Clustering by maximizing mutual information across views. In: IEEE/CVF International Conference on Computer Vision, pp 9928–9938

Edwards H, Storkey A (2017) Towards a neural statistician. In: International Conference on Learning Representations

Feifei L, Fergus R, Perona P (2006) One-shot learning of object categories. IEEE Trans Pattern Anal Mach Intell 28(4):594–611

Fink M (2005) Object classification from a single example utilizing class relevance metrics. In: International Conference on Neural Information Processing Systems, pp 449–456
Finn C, Abbeel P, Levine S (2017) Model-agnostic meta-learning for fast adaptation of deep networks. In: International Conference on Machine Learning, pp 1126–1135
Fish E, Weinbren J, Gilbert A (2021) Rethinking genre classification with fine grained semantic clustering. In: IEEE International Conference on Image Processing, pp 1274–1278
Gidaris S, Komodakis N (2018) Dynamic few-shot visual learning without forgetting. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 4367–4375
He J, Hong R, Liu X, Xu M, Zha Z, Wang M (2020) Memory-augmented relation network for few-shot learning. In: ACM International Conference on Multimedia, pp 1236–1244
Hui B, Zhu P, Hu Q, Wang Q (2019) Self-attention relation network for few-shot learning. In: IEEE International Conference on Multimedia & Expo Workshops, pp 198–203
Ji P, Zhang T, Li H, Salzmann M, Reid I (2017) Deep subspace clustering networks. In: International Conference on Neural Information Processing Systems, pp 23–32
Jiang S, Wu G (2021) Mrn: moment relation network for natural language video localization with transfer learning. Int J Pattern Recogn Artif Intell, page 2152009
Kaiser L, Nachum O, Roy A, Bengio S (2017) Learning to remember rare events. arXiv preprint arXiv:1703.03129
Kingma D, Ba J (2015) Adam: a method for stochastic optimization. In: International Conference on Learning Representations
Koch G, Zemel R, Salakhutdinov R (2015) Siamese neural networks for one-shot image recognition. In: ICML Deep Learning Workshop
Lake BM, Salakhutdinov R, Gross J, Tenenbaum JB (2011) One shot learning of simple visual concepts. In: Annual Meeting of the Cognitive Science Society
Larochelle S (2016) Optimization as a model for few-shot learning. arXiv preprint arXiv:1606.04080
Li Z, Nie F, Chang X, Nie L, Zhang H, Yang Y (2018) Rank-constrained spectral clustering with flexible embedding. IEEE Trans Neural Networks Learn Syst 29(12):6073–6082
Li Z, Nie F, Chang X, Yang Y, Zhang C, Sebe N (2018) Dynamic affinity graph construction for spectral clustering using multiple features. IEEE Trans Neural Networks Learn Syst 29(12):6323–6332
Li Z, Yao L, Chang X, Zhan K, Sun J, Zhang H (2019) Zero-shot event detection via event-adaptive concept relevance mining. Pattern Recogn 88:595–603
Liu B, Yu X, Yu A, Zhang P, Wan G, Wang R (2018) Deep few-shot learning for hyperspectral image classification. IEEE Trans Geosci Remote Sens 57(4):2290–2304
Liu Y, Schiele B, Sun Q (2020) An ensemble of epoch-wise empirical bayes for few-shot learning. In: European Conference on Computer Vision, pp 404–421
Miller EG, Matsakis NE, Viola PA (2000) Learning from one example through shared densities on transforms. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 464–471
Munkhdalai T, Yu H (2017) Meta networks. In: International Conferencerning, pp 2554–2563
Oreshkin BN, Rodriguez P, Lacoste A (2018) Tadam: Task dependent adaptive metric for improved few-shot learning. In: International Conference on Neural Information Processing Systems, pp 719–729
Qiu Z, Zhao H (2022) A fuzzy rough set approach to hierarchical feature selection based on hausdorff distance. Applied Intelligence, pp 1–14
Rahbar M, Yazdani S (2021) Historical knowledge-based mbo for global optimization problems and its application to clustering optimization. Soft Comput 25(5):3485–3501
Rusu AA, Rao D, Sygnowski J, Vinyals O, Pascanu R, Osindero S, Hadsell R (2019) Meta-learning with latent embedding optimization. In: International Conference on Learning Representations
Santoro A, Bartunov S, Botvinick M, Wierstra D, Lillicrap T (2016) Meta-learning with memory-augmented neural networks. In: International Conference on Machine Learning, pp 1842–1850
Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2020) Grad-cam: visual explanations from deep networks via gradient-based localization. Int J Comput Vision 128(2):336–359
Seo J, Yoon SW, Moon J (2020) Task-adaptive clustering for semi-supervised few-shot classification. arXiv preprint arXiv:2003.08221
Shyam P, Gupta S, Dukkipati A (2017) Attentive recurrent comparators. In: International Conference on Machine Learning, pp 3173–3181
Snell J, Swersky K, Zemel RS (2017) Prototypical networks for few-shot learning. In: International Conference on Neural Information Processing Systems, pp 4080–4090
Sung F, Yang Y, Zhang L, Xiang T, Torr PH, Hospedales TM (2018) Learning to compare: relation network for few-shot learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 1199–1208

Tang KD, Tappen MF, Sukthankar R, Lampert CH (2010) Optimizing one-shot recognition with micro-set learning. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 3027–3034
Triantafillou E, Zemel RS, Urtasun R (2017) Few-shot learning through an information retrieval lens. In: International Conference on Neural Information Processing Systems, pp 2255–2265
Vinyals O, Blundell C, Lillicrap T, Kavukcuoglu K, Wierstra D (2016) Matching networks for one shot learning. In: International Conference on Neural Information Processing Systems, pp 3637–3645
Wang X, Zhang Q, Jiang C, Zhang Y (2020) Coarse-to-fine grained image splicing localization method based on noise level inconsistency. In: International Conference on Computing, Networking and Communications, pp 79–83. IEEE
Wang Y, Hu Q, Chen H, Qian Y (2022) Uncertainty instructed multi-granularity decision for large-scale hierarchical classification. Inf Sci 586:644–661
Wang Y, Liu R, Lin D, Chen D, Li P, Hu Q, Chen CLP (2021) Coarse-to-fine: progressive knowledge transfer-based multitask convolutional neural network for intelligent large-scale fault diagnosis. In: IEEE Transactions on Neural Networks and Learning Systems, pp 1–14
Wang Z, Miao Z, Zhen X, Qiu Q (2021) Learning to learn dense gaussian processes for few-shot learning. Neural Information Processing Systems, 34
Wen W, Liu Y, Ouyang C, Lin Q, Chung T (2021) Enhanced prototypical network for few-shot relation extraction. Inf Process Manag 58(4):102596
Xing EP, Ng AY, Jordan MI, Russell SJ (2002) Distance metric learning with application to clustering with side-information. In: International Conference on Neural Information Processing Systems, pp 505–512
Xu Z, Yu F, Liu C, Wu Z, Wang H, Chen X (2022) Falcon: fine-grained feature map sparsity computing with decomposed convolutions for inference optimization. In: IEEE/CVF Winter Conference on Applications of Computer Vision, pp 350–360
Yang B, Fu X, Sidiropoulos ND, Hong M (2017) Towards k-means-friendly spaces: simultaneous deep learning and clustering. In: International Conference on Machine Learning, pp 3861–3870
Yang P, Ren S, Zhao Y, Li P (2022) Calibrating cnns for few-shot meta learning. In: IEEE/CVF Winter Conference on Applications of Computer Vision, pp 2090–2099
Yao Y (2000) Granular computing: basic issues and possible solutions. In: Joint Conference on Information Sciences, vol 1, pp 186–189
Yao Y (2004) A partition model of granular computing. Trans Rough Sets I(2):232–253
Yao Y (2011) Artificial intelligence perspectives on granular computing. In Granular Computing and Intelligent Systems, pages 17–34. Springer
Yao Y (2016) A triarchic theory of granular computing. Granular Comput 1(2):145–157
Yu X, Liu H, Wu Y, Zhang C (2021) Fine-grained similarity fusion for multi-view spectral clustering. Inf Sci 568:350–368
Zadeh LA (1997) Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. Fuzzy Sets Syst 90(2):111–127
Zhan K, Zhang C, Guan J, Wang J (2017) Graph learning for multiview clustering. IEEE Trans Cybern 48(10):2887–2895
Zhang C, Hu Q, Fu H, Zhu P, Cao X (2017) Latent multi-view subspace clustering. In: IEEE Conference on Computer Vision and Pattern Recognition, pp 4279–4287
Zhang H, Zhu J, Chen J, Liu J, Ji L (2021) Zero-shot fine-grained entity typing in information security based on ontology. Knowl-Based Syst 232:107472
Zhang R, Song X, Ying S, Ren H, Zhang B, Wang H (2021) Ca-csm: a novel clustering algorithm based on cluster center selection model. Soft Comput 25(13):8015–8033
Zhao H, Hu Q, Zhu P, Wang Y, Wang P (2021) A recursive regularization based feature selection framework for hierarchical classification. IEEE Trans Knowl Data Eng 33(7):2833–2846
Zhao H, Wang P, Hu Q, Zhu P (2019) Fuzzy rough set based feature selection for large-scale hierarchical classification. IEEE Trans Fuzzy Syst 27(10):1891–1903
Zheng Q, Zhu J, Li Z, Pang S, Wang J, Li Y (2020) Feature concatenation multi-view subspace clustering. Neurocomputing 379:89–102
Zhou S, Deng C, Piao Z, Zhao B (2020) Few-shot traffic sign recognition with clustering inductive bias and random neural network. Pattern Recogn 100:107160