Applied Intelligence
https://doi.org/10.1007/s10489-022-03332-7

Few-shot learning via relation network based on coarse-grained granulation

Xiao Jia[1,2] · Yuling Su[1,2] · Hong Zhao[1,2]

Accepted: 1 February 2022

Abstract

Few-shot learning, which aims to identify new classes with very few samples, is an increasingly popular and crucial research topic in the machine learning. Many models use distance measurement to determine similarities among single samples and achieve accurate classification results. However, distance calculations incur substantial costs and time based on a single sample, and the linear measurement model cannot accurately represent the differences and connections between samples. This paper proposes a coarse-grained granulation relation network (CGRN) model for few-shot classification. First, all the single samples of each class are clustered into coarse grain to represent the feature information of all the class samples, which can significantly reduce computational complexities. Second, a relation network is built to measure the degree of similarity among the test samples and the coarse grain obtained above, which can reveal the differences and connections between the samples. The experimental results demonstrate that this model outperforms some popular distance measurement-based few-shot learning models. For example, CGRN is at least 0.5% better than other models in *20-way 5-shot* on the *Omniglot* dataset and achieves 0.8% improvement over the second-best model in *5-way 1-shot* on the *tiered-ImageNet* dataset.

Keywords Granular computing · Machine learning · Few-shot learning · Image classification

1 Introduction

Few-shot learning [24, 30] is a special application scenario of machine learning [2] that mainly addresses problems such as huge demands for deep learning data [12, 14], high costs of manual labeling, uneven data distribution, rare number of samples, and the continuous emergence of new samples. Recent years have witnessed an increased use of few-shot learning [17, 31] in various domains such as behavior detection, language translation, video classification, audio classification, and rare animal protection [32, 33].

Similarity metric learning is one of the principal models in few-shot learning [1, 5]. For instance, Edward et al. [3] created a separate variational auto-encoder to share factor coding and successfully measured different similarities among samples in various aspects. Koch et al. [10] adopted a similar and unique convolution structure to arrange the similarity between inputs naturally and completed the classification by calculating the weighted component-level distance. In contrast to previous models, some researchers used cosine similarity to measure the similarity among samples. Vinyals et al. [29] used the latest advances in attention and memory mechanisms and implemented similarity measurement by embedding the cosine distance. Furthermore, Santoro et al. [23] built a memory-enhanced neural network using cosine similarity measures to recognize new data quickly. With further improved cosine similarity, Kaiser et al. [7] adopted the end-to-end training model and used cosine similarity ranking to determine the nearest neighbor to prediction.

Meta-learning is the second-most primary model among few-shot learning models [18, 28]. For example, Sun et al. [27] applied meta-learning to learn the depth distance measure for image classification. Similarly, Finn et al. [4] constructed a model-agnostic meta-learning model that used distance as the reward function of the model parameter gradient update. In contrast to the previous two models, Munkhdalai et al. [15] learned the meta-level knowledge

✉ Hong Zhao
hongzhaocn@163.com

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

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

of cross-task learning by constructing the meta-learning network and used the layer enhancement technology for classification. By adding the memory function to meta-learning, Ravi et al. [19] proposed a meta-learning algorithm based on the long short-term memory that obtained the basic knowledge shared by all tasks.

Most of the abovementioned models use distance measurement for measuring the similarity among single samples and accomplish effective results [8, 25]. However, calculating the distance between the test sample and every single sample is inefficient. Furthermore, distance expressed as a linear value does not account for more complex relationships [6, 13].

This paper proposes a few-shot learning model for a relation network based on coarse-grained granulation (CGRN). This model applies coarse-grained granulation to the classes and introduces a measurement module for matching the similarity. First, feature information of each class was extracted for coarse-grained granulation. The coarse grain with the highest matching degree was only needed to classify the embedded query sample. This action can significantly reduce the computational complexity while maintaining the simplicity and efficiency of the model. Second, the relation network was used to construct a measurement module and calculate the similarity between coarse grain and embedded test sample to obtain the relation scores. The relation score can not only reflect the distance between samples, but also better reflect the differences and connections between samples. Finally, the prediction classes of the embedded test samples were obtained by comparing the scores of embedded test samples with those of all coarse grain relation.

The experiments were mainly conducted on the *Omniglot*, *tiered-ImageNet*, and *mini-ImageNet* datasets. CGRN has established its superiority over other comparable models in most experiments. Especially, it is better than other models by at least 0.5% in the *20-way 5-shot* experiment on the *Omniglot* dataset. CGRN outperforms OVE PG G P+Cosine by about 0.8% [26] in the *5-way 1-shot* experiment on the *mini-ImageNet* dataset. In the *5-way 5-shot* setting, our model is comparable with other models. This model outperforms other models by at least 0.8% in a *5-way 1-shot* setting in the experiments conducted on the *tiered-ImageNet* dataset.

Our main contributions can be summarized as follows:

(1) A coarse-grained granulation is proposed to represent the characteristic information in a class. Unlike classifying by a single sample, the sample features of every class are granulated to obtain the more generalized class features.
(2) A relation module is leveraged to measure the image similarity instead of the distance metric, which can better reflect the class relationship and promote the model classification ability.

The remainder of this paper is organized as follows. Section 2 presents the details of the proposed model. Next, Section 3 presents a detailed explanation of the experimental setup. Then, Section 4 introduces and analyzes the experimental results. Finally, Section 5 presents conclusions and inspirations for future works.

2 Relation network based on coarse-grained granulation

This section provides a detailed description of the structure of the relation network based on coarse-grained granulation (CGRN).

2.1 Framework

Our model is mainly divided into two parts: coarse-grained granulation and relation module. Figure 1 shows the CGRN framework.

(1) The coarse-grained granulation part mainly deals with the support data for creating the coarse grain that represents the fundamental information of each class's samples.
(2) The relation module calculates relation scores by splicing the coarse grain and query samples. Then, the classification results are obtained by comparing the relation scores.

2.2 Coarse-grained granulation

The first part of the model is coarse-grained granulation for each class. In each iteration of training, the C classes are randomly selected from the training set and K samples of each class are used as the support set $S = \{x_i, y_i\}_{i=1}^{m}$, where $m = CK$ is the number of samples. Then, the remaining part of the sample in class C is taken as the query set $Q = \{(x_j, y_j)\}_{j=1}^{n}$, where n is the number of samples in the query set.

x_i is assumed as the sample in the support set S, $y_i \in \{1, ..., C\}$ is taken as its label, x_j is the sample in the query set Q, and y_j is the corresponding label. The samples are sent into the embedded module f_φ of CGRN. The convolutional neural network extracts the samples for special diagnosis, obtaining the feature map $f_\varphi(x_i)$ with the sample attributes. For instance, most famous learning models use four convolution blocks to embed the model [25, 29]. Specifically, each convolution block comprises a 64-filter 3×3 convolution, a batch normalization layer, and a $ReLU$ nonlinear layer. The first two convolution blocks

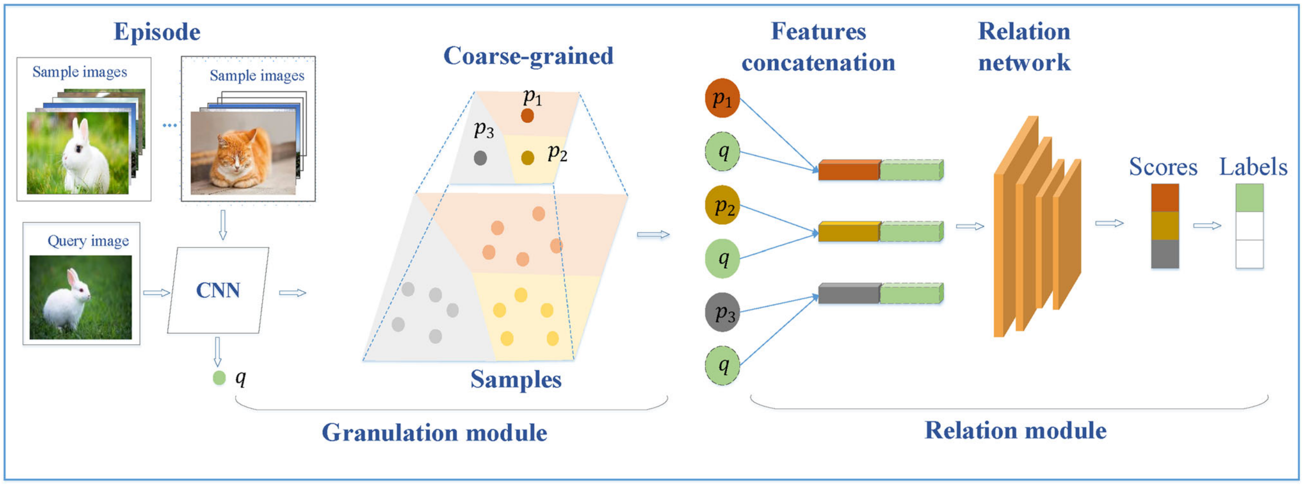


Fig. 1 The CGRN framework. Parameters p_1, p_2, and p_3 are three coarse-grains and q represents a query feature

also contain a 2×2 maximum pooling layer, whereas the last two do not contain the same. This arrangement was made because the output feature graph was used for other convolution layers in the relation module.

In the classification process, all samples are assumed to be in the same sample space and learn embedding in the metric space at sample positions. First, coarse-grained granulation with every class was constructed in the support set. A central point is designated as the representative point of all samples when every type is in the same sample space. This point should contain the basic information of this class's samples and the majority of the sample attributes. It can represent all the samples to compare with the query samples during classification. The center point is the coarse grain (p_c), which is obtained by granulation. Each coarse grain is the mean vector of feature mapping of embedded sample points belonging to this class, and the calculation formula is shown as follows:

$$p_c = \frac{1}{|S_c|} \sum_{(x_i, y_i) \in S_c} f_\varphi(x_i), \tag{1}$$

where S_c denotes the set of examples labeled with class C, $|\cdot|$ is the number of the samples in S_c. The coarse grain can represent the basic attributes of the classes, covering the basic attributes of the class compared with other learning models. The structure of the coarse-grained representation is simple, practical, and effective, which can well improve the effectiveness of classification and simplify the model complexities. Next, Example 1 compares the difference between distance and coarse-grained similarity measurements.

Example 1 Figure 2 shows different situations in which the support set contains several images of different species.

In the distance measurement process, a distance value is calculated between all the images in the sample set and the test sample. These distance values are d_1, d_2, and d_3, as shown in Fig. 2. Figure 2a shows the distance d_2 between the new sample *black cat* and *orange cat*; however, there are two *orange cat* pictures. It is essential to calculate the distance of multiple error classes in this scenario.

Therefore, each class obtains the coarse grain, which represents each class. As shown in Fig. 2b, the mean value of sample pictures 1, 2, 3, 4, 5, 6, and 7 is calculated as the coarse-grained points. In this way, three class coarse-grained points, which are used as the representations of the three classes to match with the test samples, were obtained. In this scenario, it is vital to calculate the distance value d_1, d_2 and d_3 between the three coarse grains and the test sample to predict its class. Compared with the traditional models, our model is more efficient because it needs a small number of computations to obtain the classification results.

2.3 Relation network based on coarse-grained granulation for few-shot classification

Most of the few-shot learning models generally adopt distance measurement to perform similarity matching classification after obtaining the sample features from the support and test sets. Many distance functions such as Euclidean distance, Mahalanobis distance, Manhattan distance and so on are currently available for different applications. The current research works prove that the distance functions are helpful in sample similarity matching. However, the distance value obtained through the distance function can only reflect the far and near relations between the two samples. The distance value is just a linear value as a similarity measurement unit. Example 2 describes the shortcomings of distance similarity measurement.

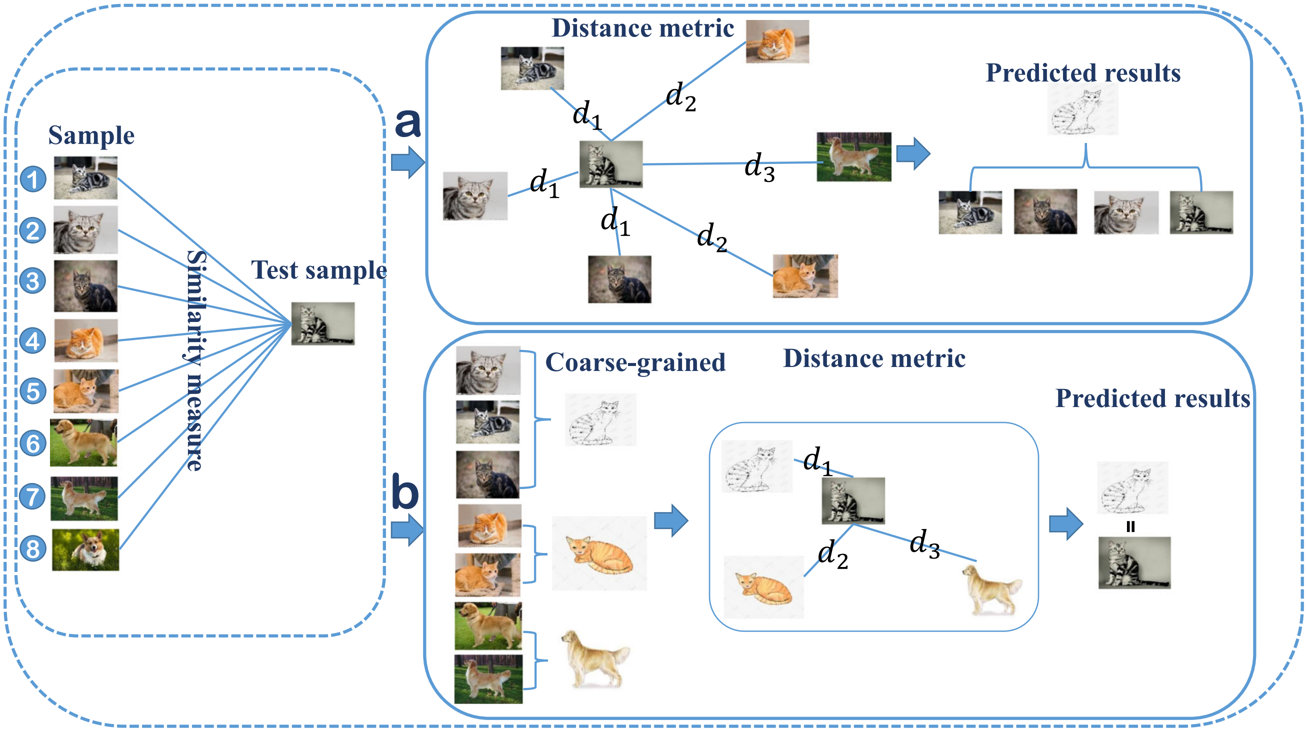
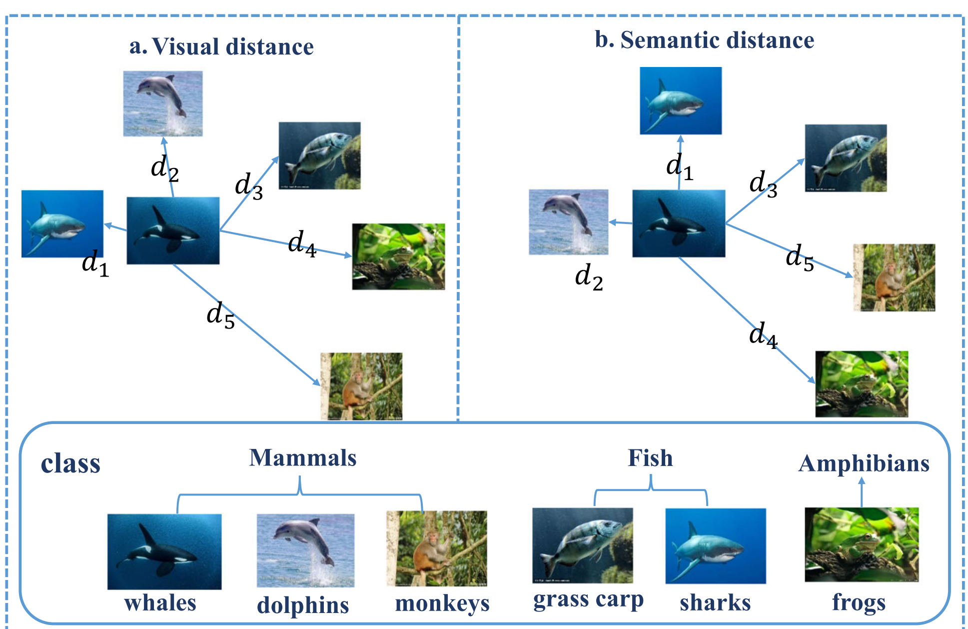


Fig. 2 Coarse-grained granulation. Parameters d_1, d_2, and d_3 are three distance values

Example 2 Figure 3 shows six samples, namely *whales, sharks, dolphins, grass carp, frogs* and *monkeys*, in the distance similarity metric space. They belong to the three classes of *mammals, fish* and *amphibians*. The following observations can be made: (1) In Fig. 3a, the d_1 is the smallest because *whales* and *sharks* are very similar in terms of visual features. (2) In Fig. 3b, the distance d_2 is the smallest because *whales* and *dolphins* belong to the class of

Fig. 3 Disadvantages of the distance calculation model. Parameters d_1, d_2, d_3, d_4, and d_5 are five distance values



mammals. Both use distance to calculate the similarity, but the results are inconsistent [34, 35].

Distance is a linear value in the classification of distance from different angles. There is a distance relation between the two samples and an intricate relation between the classes they belong to. These relations can also be used as auxiliary information to the classification task for guiding the classification task. Our model uses relation score as a unit of measurement for the similarity measure.

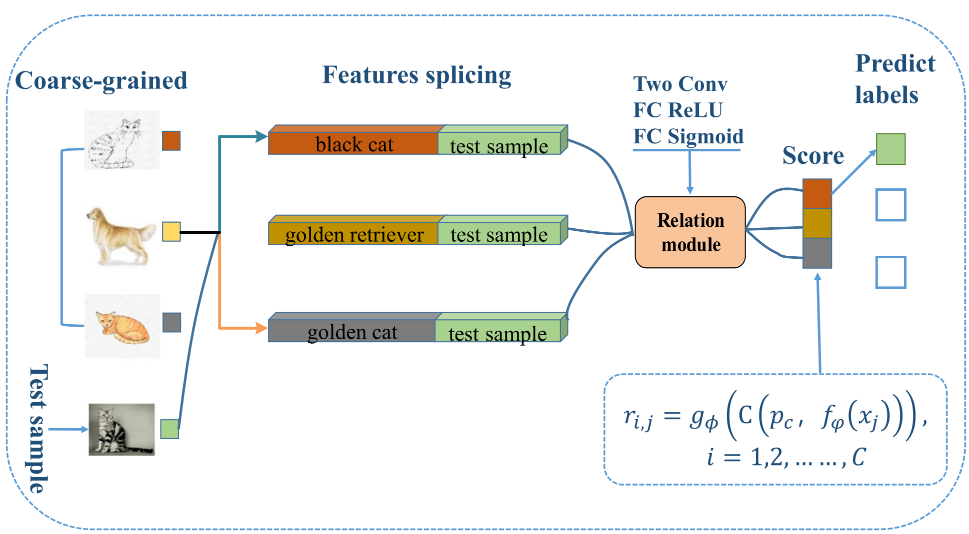
In the first step of the model, the class coarse grain of each class is obtained by embedding the module and the feature mapping of the coarse grain is extracted through the neural network f_φ. In the subsequent step, this class coarse grain is used to classify the query samples. The distance value between each coarse grain in each batch and the query sample in the original coarse-grained network is calculated via the distance function to represent their relation. The query sample x_j is sent into the embedded module to obtain the feature map $f_\varphi(x_j)$. Then, the sample set coarse grain p_c and the feature map $f_\varphi(x_j)$ are combined with the operator $\mathcal{C}(p_c, f_\varphi(x_j))$. $\mathcal{C}(\cdot, \cdot)$ is assumed to concatenate feature mappings at depth; however, there are other ways to perform this action. Then, the combined feature mapping is sent to the relation module g_ϕ for calculation. The relation module consists of two convolution layers and full connection layers. Each convolution block is a 3×3 convolution with 64 filters, followed by batch normalization, *ReLU* nonlinearity, and 2×2 maximum pooling. All the full connection layers are *ReLU* and the output layer is Sigmoid, which enables the network architecture to produce a reasonable range of results. Figure 4 shows the relation module.

The final output result of the relation module is a scalar within the range of 0-1, representing the matching degree between p_c and x_j. The relation score $r_{i,j}$ is computed as follows:

$$r_{i,j} = g_\phi(\mathcal{C}(p_c, f_\varphi(x_j))), \quad i = 1, 2, ..., C, \tag{2}$$

where $f_\varphi(x_j)$ is the feature map of x_j, $\mathcal{C}(p_c, f_\varphi(x_j))$ is combined by the sample set coarse grain p_c and feature map $f_\varphi(x_j)$. The relation score reflects the relation between two samples. The higher the relation score, the higher the matching degree between the coarse grain and the test sample. There is an increased possibility of test samples belonging to this class. The support set contains multiple classes, each of which can form the coarse grain of the class. The relation score between the test sample and each coarse grain can be obtained during calculation, and the most likely class of the test sample can be predicted by comparing all the obtained relation scores. In the *C-way K-shot*, a sample set coarse grain is obtained for each class in the embedded module, and the sample set selected in *C-way One-shot* is the coarse grain for this class. For the objective function, the mean square error loss is used to train our model and return $r_{i,j}$ to the basic fact. The similarity is 1 when the classification is correct. Moreover, the similarity is 0 when the classification is wrong. The objective function is calculated as follows:

Fig. 4 Relation module



Algorithm 1 Few-shot learning via relation network based on coarse-grained granulation (CGRN).

Input: Sample set $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{m}$ $(m = KC)$, where C means that C classes are randomly selected from the training set, and K means that each class contains K labeled samples. A portion of the remaining sample in C classes is taken as query set $\mathcal{Q} = \{(x_j, y_j)\}_{j=1}^{n}$.
Output: The loss $\mathcal{L}$ for a randomly generated training episode.

```
1:  The sample of the training set and its label are obtained
    from the sample set S = {(x_1, y_1), ..., (x_n, y_n)}, S_c is
    the sample set labeled with class C;
2:  Extract features of Q and S by feature extracting
    module;
3:  for i = 1 : m do
4:      for j = 1 : n do
5:          Compute coarse-grained classes p_c by (1);
6:          Compute classification scores r_{i,j} by (2);
7:      end for
8:  end for
9:  Initialize loss L = 0;
10: for i = 1 : m do
11:     for j = 1 : n do
12:         Update classification loss L by (3);
13:     end for
14: end for
```

$$\varphi, \phi \leftarrow \arg\min \sum_{i=1}^{m} \sum_{j=1}^{n} (r_{i,j} - \mathbf{1}(y_i == y_j))^2, \tag{3}$$

where $r_{i,j}$ is the relation score of matching degree between p_c and x_j. The choice of the *MSE* function does not seem to make much sense for a classification problem where the tag space is $\{0, 1\}$. However, conceptually, the relation scores are predicted, which can be considered as a regression problem. Algorithm 1 provides the training model and pseudocode for a training episode. In particular, the coarse grain and the classification score listed from lines 3-8 are computed, and the classification loss is calculated in lines 10-14.

Two popular few-shot learning models inspired our model: relation network [27] and prototype network [25]. Similar network structure and experimental settings were adopted in this model. The classification is completed by calculating the coarse-grained granulation and relationship score, which simplify the model to a certain extent. In terms of time complexity, our model is at the same level as the few-shot learning models based on the relation network and prototype network.

3 Experimental settings

This section introduces the experiments of our model in detail. The experimental introduction mainly includes datasets, comparison models, and parameters.

3.1 Datasets

Our proposed relation network based on coarse-grained granulation is mainly carried out on the *Omniglot*, *mini-ImageNet*, and *tiered-ImageNet* datasets. The three datasets have a wide range of applications in the field of few-shot learning and meta-learning.

(1) *Omniglot* [11] contains 1623 different handwritten characters with 50 different letters, and every picture in the dataset was drawn by 20 different people using Amazon's Mechanical Turk. This corresponds to 1623 classes in the *Omniglot* handwritten dataset, each of which consists of 20 samples.
(2) *mini-ImageNet* is an excerpt from the ImageNet dataset. The *mini-ImageNet* dataset consists of 100 classes, each class consists of 600 samples and a total of 60,000 color slides.
(3) *tiered-ImageNet* [21] is a subset of the ILSVRC-12 [22] dataset containing more than 700,000 images. According to the class hierarchical structure of *ImageNet*, there are 608 fine-grained classes and 34 coarse-grained classes.

3.2 Comparative models

In our experiments, we compare our model with many popular few-shot learning models. These few-shot learning models are briefly introduced as follows.

(1) Relation Net [27]: In the relation network, the model is divided into two parts: feature extraction model and embedded module.
(2) Prototype Net [25]: Class prototype is obtained for each class in the embedded space, and then the distance between the query samples and these prototypes is calculated to classification.
(3) Neural Statistician models (NS) [3]: By extending the variable auto-encoder, the model realizes the unsupervised computation of dataset representation.
(4) Matching Nets [29]: The neural memory enhancement network based on the attention mechanism and the memory mechanism is used to carry out similarity matching between the two samples for classification.

(5) MANN [23]: The meta-learning model of memory-enhanced neural network improves the traditional focusing mechanism based on memory location.
(6) Siamese Nets with Memory (SNM) [7]: The Siam Net approach with memory is an end-to-end life-long memory module that can be added to any part of the supervised neural network for classification.
(7) Convolutional Siamese Nets (CSN) [10]: In the Siamese Net model of convolution, a model of learning Siamese neural network is discussed, which uses a unique convolutional structure to arrange the similarity between the inputs naturally.
(8) Model unknowable meta-learning (MAML) [4]: Model unknowable meta-learning model is a model to train model parameters explicitly but easily fine-tune, which can quickly adapt to different depth neural networks.
(9) Meta Nets [15]: The learning model of meta-network uses a layer enhancement technique to classify by weight information of different time scales and class information of external memory modules.
(10) Meta-learning algorithms for long and short-term memory networks (Meta-Learner LSTM) [19]: The algorithm can capture the basic knowledge shared among all tasks and parameterize the model so that it can learn the appropriate parameter update.
(11) BOIL [16]: The meta-learning algorithm updates only the body (extractor) of the model and freezes the head (classifier) during inner loop updates. It visualizes this property using cosine similarity.
(12) OVE PG G P+ Cosine (ML) [26]: It proposes a Gaussian process classifier based on a novel combination of augmentation and the one-vs-each softmax approximation.
(13) ECMT [20]: It is a model for learning embedding for few-shot learning that is suitable for use with any number of shots (*shot-free*).

3.3 Experimental setup

The experiments were conducted on a desktop computer running Ubuntu 18.04 with 31.0 *GB* of RAM, an Intel Xeon (R) W-2133CPU@3.60GHZ12 CPU, and a GeForce RTX 2080TI/PCIe/SSE2 graphics card. Adam [9] is used in all the model experiments. All experiments were performed from scratch and randomly initialized without incorporating any other dataset. The initial learning rate is 0.001, and half of annealed is repeated for every 10,000 batches. The *C-way K-shot* setting was used for all the experiments. If the training set has C unique classes and each class has K labeled samples, then the target few-shot problem is *C-way K-shot*.

In accordance with the standard settings adopted by most few-shot learning models, 5 classes and 20 classes were successively selected from the training set in the experiment, with 1 and 5 samples selected from each class, respectively. In experiments on the *Omniglot* dataset, except for the K sample images, the *5-way 1-shot* contains 19 query images, the *5-way 5-shot* contains 15 query images, the *20-way 1-shot* contains 10 query images, and the *20-way 5-shot* contains 5 query images for C sample classes in the training set. This implies that each training set/small batch in the *5-way 1-shot* experiment has $19 \times 5 + 1 \times 5 = 100$ images. Each training set/small batch in the *5-way 5-shot* experiment has $15 \times 5 + 5 \times 5 = 100$ images. Each training set/small lot in the *20-way 1-shot* experiment has $10 \times 20 + 1 \times 20 = 220$ images. Each training set/small batch in the *20-way 5-shot* experiment has $5 \times 20 + 5 \times 20 = 200$ images. All image sizes in the dataset are set to 28×28 to facilitate subsequent network calculations. In experiments on the *mini-ImageNet* and *tiered-ImageNet* datasets, except for K sample images, *5-way 1-shot* contains 15 query images, and *5-way 5-shot* contains 10 query images, which are used for C sample classes in the training set. This means that each training set/small batch in the *5-way 1-shot* experiment has $15 \times 5 + 1 \times 5 = 80$ images. Each training set/small batch in the *5-way 5-shot* experiment has $10 \times 5 + 5 \times 5 = 75$ images.

4 Experimental results and analysis

4.1 Performance using different K-shot samples

The number of samples is critical for model training. Different numbers of labeled samples were selected in each class for training models.

First, five classes were randomly selected in the training set on the *Omniglot* dataset. Then, one sample was selected in each class. Except for K sample images in the experiment of the *Omniglot* handwritten dataset, *5-way 1-shot* contains 19 query images, with each training set/small batch containing $19 \times 5 + 1 \times 5 = 100$ images. This sample is the coarse grain of this class and comprises basic information of classes when one sample is selected from each class for training in this embedded space. In the test process, the coarse grain of this class will perform similarity matching with the test sample and calculate accuracy according to the matching results. In the same setting, experiments with 5, 10, and 15 samples of the support set were also conducted. The query samples are 15, 10, and 5, respectively. The same setup is used in the experiment on the *Omniglot* dataset *20-way* to establish the validity of our model.

Table 1 shows the accuracy of our model on the *Omniglot* dataset in *5-way* cases. The classification accuracy changes of the model can be determined in different ways, which show a trend of rising first and falling afterward. The results present the following observations:

(1) The classification accuracy of *5-way 1-shot* is lowest, whereas the classification accuracy of *5-way 5-shot* is improved by 0.4% and reaches the highest point of the results of this group of experiments. The basic information of class is provided by one sample is insufficient. The obtained coarse grain cannot accurately represent the whole class with low classification accuracy.
(2) In the case of the *5-way* experiment, our model achieves an accuracy of 99.78%, an improvement of 0.4%. It is the most reasonable to take five samples from each class, and the classification result of querying samples is also the highest. The coarse grain of each class can obtain all the class information when there are five samples and generate the coarse grain of classes to complete the classification.
(3) In the cases of *5-way 10-shot* and *5-way 15-shot* settings, the classification accuracy of the experiment was decreased by 0.1% and remained stable at 99.84%. It indicates that the classification efficiency of our model has reached the highest level in the *5-way 5-shot* setting. Excessive sample information acquired in generating the coarse grain will produce redundancy, interfering with the classification results. The classification accuracy is reduced when the number of samples of each class is more than 5, such as 10 and 15. Too much useless information affects the classification results when the coarse grain obtained contains all the class information.

Table 2 shows the experimental results of our model in the cases of *20-way* on the *Omniglot* dataset. By comparing the above four situations, 20 random classes and different samples were selected from each class to generate the coarse grain. Table 2 shows that the classification accuracy of the model changes under different situations, demonstrating an increasing trend on the whole. The results show the following observations:

(1) The classification accuracy is the lowest in the case of *20-way 1-shot*, and the classification accuracy is constantly improved with the increase in the number of samples of each class. When the number of samples in each class is increased from one to five, the classification ability of the model is greatly improved by 2%, which is very significant at *20-way* with an average accuracy of 98.7%. This outcome indicates that the basic information is obtained when there is only one sample for each class is insufficient and the classification cannot be completed properly.
(2) The classification accuracy continues to improve, increasing by 0.1% in the *20-way 10-shot* and *20-way 15-shot* cases, respectively, with the increase in the number of sample classes. This result indicates that good coarse grain can be obtained with five samples in each class. The improvement in the model is not apparent when the number of samples for each class is too large.
(3) Furthermore, the training time in the case of *20-way* is geometrically increased than that in the *5-way* setting, which significantly increases the resources required to train the model.

Table 1 Performances comparison with different *K-shot* samples of *5-way* (%)

K-shot	1-shot	5-shot	10-shot	15-shot
Acc	99.57	99.85	99.84	99.84

Table 2 Performances comparison with different *K-shot* samples of *20-way* (%)

K-shot	1-shot	5-shot	10-shot	15-shot
Acc	97.25	99.09	99.21	99.31

In the cases of *20-way*, the time required for *1-shot*, *5-shot*, *10-shot*, and *15-shot* training models shows an increasing linear state. Considering the loss of resources and the variation of model classification accuracy, the model can be obtained by selecting 1 and 5 samples from each class in our *20-way* experiments.

4.2 Performance comparison of models using Top *k* relation scores

In this section, experiments were conducted using Top k relation scores to validate the effects of our model on the *Omniglot* dataset. The class coarse grain and test samples were matched by similarity, and the accuracy was calculated according to the matching results. The relation score was used to measure the similarity between the two samples. The relationship score of Top k was used to reflect the change of labels predicted by our model.

Table 3 lists the classification accuracy of our model by selecting different relation scores. The following observations may be drawn from the results:

(1) Our model achieves an accuracy of 97.25% when the class with the highest relation score is used as the prediction class. The model performance continuously

Table 3 Performances comparison with different relation scores of *20-way 1-shot* (%)

Top k	Top 1	Top 2	Top 3	Top 4
Acc	97.25	99.35	99.67	99.81

improves as the rules for selecting the relation score change.

(2) The classification accuracy is improved by 2.1% when the two prediction classes with the highest relation score are used. Therefore, the two prediction classes can be comprehensively considered while making prediction classes for a sample. The sample prediction is considered accurate when one of the two prediction classes is correct. This model can improve the prediction accuracy of samples.

(3) The prediction accuracy is increased by 0.3% when the selection of relation score increases from Top 2 to Top 3. The prediction accuracy is increased by 0.2% when the selection of relation score increases from Top 3 to Top 4. The prediction classes of the first several results with the most significant relation scores are similar. It proves that the classification ability of our model has a better effect. The classification of test samples conforms to the real results, and the residual errors are influenced mainly by randomness.

Example 3 provides an intuitive understanding of the relationship between relation scores and prediction labels. In this example, a test set that includes ten query samples, randomly selected from a batch of 20 types of samples, is used. Table 4 presents the true labels and different prediction labels of these ten samples.

Example 3 Table 4 shows that the prediction accuracy of our model reaches 100% when the relation score Max is taken and the predicted label and its true label of each sample are the same. Then two prediction labels are obtained when the relation score is Top 2. For example, the true label of *sample* 1 is (13), and the predicted labels of *sample* 1 is (15, 13) when the Top 2 relation score is used to predict the labels. Similarly, Table 4 shows the prediction labels of Top 3 and Top 4 samples. For example, the prediction labels of *Sample* 1 are (15, 6, 13) and (19, 6, 15, 13), respectively. Our model achieves higher accuracy in the prediction of labels.

Table 5 Performances comparison with different relation scores of *5-way 5-shot* (%)

Top k	Top 1	Top 2
Acc	64.13	85.13

Similarly, these experiments were conducted on the *mini-ImageNet* dataset. The changes of the prediction accuracy and the deviation of the possible results while classifying a test sample are determined by selecting the output prediction label under different relation scores in the *5-way 5-shot* setup. Table 5 shows the accuracy rate under different quantitative relation scores.

Table 5 shows increasing relation score choices and improved accuracy of classification. The classification accuracy is increased from 64.13% to 85.11%. According to the data, the label prediction of the real class of the sample by our model is relatively accurate and close to its real label.

Example 4 provides an intuitive understanding of the relationship between relation scores and prediction labels. Table 6 enlists the true labels and different prediction labels of these ten samples.

Example 4 Table 6 shows the changes in the possibility of predicting labels in these ten samples. For example, *sample* 1 has true label 0. The prediction label obtained is 1, which

Table 4 True label and prediction label of ten samples

Sample	1	2	3	4	5
True label	13	6	12	1	3
Top 1	13	6	12	1	3
Top 2	(15, 13)	(8, 6)	(1, 12)	(11, 1)	(17, 3)
Top 3	(15, 6, 13)	(8, 19, 6)	(4, 1, 12)	(11, 12, 1)	(6, 17, 3)
Top 4	(19, 6, 15, 13)	(2, 17, 8, 6)	(11, 18, 1, 12)	(18, 11, 12, 1)	(15, 17, 11, 3)
Sample	6	7	8	9	10
True label	11	9	15	5	7
Top 1	11	9	15	5	7
Top 2	(10, 11)	(10, 9)	(19, 15)	(0, 5)	(15, 7)
Top 3	(18, 10, 11)	(18, 10, 9)	(13, 19, 15)	(4, 0, 5)	(3, 15, 7)
Top 4	(14, 1, 10, 11)	(3, 18, 10, 9)	(4, 8, 6, 15)	(11, 0, 4, 5)	(17, 15, 6, 7)

Table 6 True labels and predicted labels of ten samples

Samples	1	2	3	4	5	6	7	8	9	10
True labels	0	1	2	3	4	0	1	2	3	4
Top 1	1	1	4	3	4	0	4	1	3	4
Top 2	(0, 1)	(3, 1)	(0, 4)	(1, 3)	(1, 4)	(2, 0)	(3, 4)	(2, 1)	(2, 3)	(2, 4)

is wrong when the selection relation score is Top 1. The prediction labels are (0,1) with the Top 2 relation scores. In this case, the prediction is considered correct if the true label is present within the prediction labels. The prediction accuracy of Top 2 reaches 85.13%, which increases the accuracy by 20% than that of the one prediction label. The performance of Top 2 is far exceeding the average level of the *5-way 5-shot* experiment in the general *mini-ImageNet* dataset.

4.3 Performance comparison of model training times

This section presents a series of experiments conducted on the number of iterations of our model training. Hence, the influence of the best number of iterations was verified in the experiment. Five random classes were randomly selected on the *Omniglot* handwritten dataset, and one sample was selected from each class for training. Except for the K sample images on the *Omniglot* dataset experiment, *5-way 1-shot* contains 19 query images and each training set/small batch has $19 \times 5 + 1 \times 5 = 100$ images. This sample is the coarse grain of this class and contains all the basic information of this class when one sample was selected from each class for training in this embedded space. The number of model training was successively increased from 1×10^5 times including 5×10^5, 10×10^5, and 20×10^5 to verify the influence of the number of iterations of model training. The optimal model training time is found through different iterations of the model, which is used as the benchmark for the further training of the model.

Table 7 shows the experimental results under different iterations. From Table 7, the following observations were noted:

(1) The parameters provided by the training model are insufficient when the number of iterations in the model are 1×10^5 and 5×10^5, and the obtained model is not the best model. Therefore, the prediction accuracy of the test samples is not the highest. The accuracies were 98.95% and 99.36%. The model cannot be trained to be well enough to realize classification.

(2) The performance of our model is best when the training times are 10×10^5 times. The accuracy of the test samples is improved to 99.57%, which reaches the highest at 965,000 times. This outcome indicates that in the subsequent comparative experiments, the optimal number of iterations in our experiments should be set at 10×10^5.

(3) However, when the number of iterations in the model increases to 20×10^5 again, the test accuracy of the model does not increase with the increase in the number of iterations, but is decreased by 0.1% to 99.44%. The over-fitting phenomenon emerges when the number of iterations is too high and reduces the model accuracy. Simultaneously, this large calculation process brings much loss to the machine and wastes much time.

Table 7 Performances of *5-way 1-shot* convergence analysis (%)

Times	1×10^5	5×10^5	10×10^5	20×10^5
Acc	98.95	99.36	99.57	99.44

4.4 Ablation studies

This section shows that both modules of the relation and prototype nets facilitate the proposed model performance. The experiments were mainly conducted on the *Omniglot* and *mini-ImageNet* datasets. Table 8 presents the experimental results, which convey the following observations:

(1) Our model is comparable with the relation net of classification via sample by sample in most cases. In the experiments of the *5-way 5-shot* on the *Omniglot* and *5-way 1-shot* on the *mini-ImageNet*, our model

Table 8 Effectiveness of different modules on Omniglot and mini ImageNet (%). The best performing model is highlighted

Dataset	Prototype net	Relation net	*5-way 1-shot*	*5-way 5-shot*
Omniglot	✓		98.80	99.70
		✓	99.55	99.74
	✓	✓	**99.57**	**99.85**
mini-ImageNet	✓		49.42 ± 0.78	**68.20 ± 0.66**
		✓	50.44 ± 0.82	65.32 ± 0.70
	✓	✓	**50.85 ± 0.86**	64.13 ± 0.70

goes beyond the relation network model. In particular, this model obtains about 0.4% improvement over others. Our model is slightly insufficient in the *5-way 5-shot* experiment of the *mini-ImageNet* dataset. It is believed that the main reason is that the dataset has complex sample types and a wide range of features, which affect the efficiency of coarse-grained granulation.

(2) Table 8 demonstrates that the model has significant advantages over the prototype net relying on distance computing. On the *Omniglot* dataset *5-way 1-shot*, this model is about 0.7% better than the prototype and achieves an improvement of 1.4% over it on the *mini-ImageNet* dataset in the *5-way 1-shot* setting. Our model is slightly insufficient on the *5-way 5-shot mini-ImageNet* dataset. The main reason is the presence of too many classes in the dataset, which can lead to misclassification.

4.5 Performance comparison with different models

Finally, our model is compared with some other few-shot learning models. Compared with these models, this model superiority can be found in the few-shot learning model in recent years. The best-performing model is highlighted, and '-' indicates that the accuracy of the model is not reported.

Table 9 presents the performance comparison between our model with different models on the *Omniglot* dataset. Most of them adopt the distance measurement strategy. For example, MANN, CSN, and Matching Nets adopt distance measurement and take the distance value between two samples as the main basis for classification. According to the results, the following observations can be made: our model is better than most models based on the distance measurement model, especially in the *20-way 5-shot* experiment when the other models reach a maximum of 98.10% and the improvement is more than 1%. The classification results of the *Omniglot* dataset show the effectiveness of our model.

Table 9 Performance comparison between our model with different models on the Omniglot dataset (%). The best performing model is highlighted

Model	Fine	5-way Acc		20-way Acc	
		1-shot	5-shot	1-shot	5-shot
MANN [23]	N	82.80	94.90	–	–
CSN [10]	N	96.70	98.40	88.00	96.50
CSN [10]	Y	97.30	98.40	88.10	97.00
Matching net [29]	N	98.10	98.90	93.80	98.50
Matching net [29]	Y	97.90	98.70	93.50	98.70
SNM [7]	N	98.40	99.60	95.00	98.60
NS [3]	N	98.10	99.50	93.20	98.10
Meta net [15]	N	99.00	–	97.00	–
CGRN	N	**99.57**	**99.85**	**97.25**	**99.09**

Table 10 Performance comparison between our model with different models on the mini-ImageNet dataset (%). The best performing model is highlighted

Model	Fine	5-way Acc	
		1-shot	5-shot
Meta-Learn LSTM [19]	N	43.44 ± 0.77	60.60 ± 0.71
Matching net [29]	N	43.56 ± 0.84	55.31 ± 0.73
MAML [4]	Y	48.70 ± 1.84	63.11 ± 0.92
ECMT [20]	N	49.07 ± 0.43	**65.73 ± 0.36**
Meta net [15]	N	49.21 ± 0.96	–
BOIL [16]	N	49.61 ± 0.16	66.45 ± 0.37
OVE PG G P+ Cosine (ML) [26]	N	50.02 ± 0.35	64.58 ± 0.31
CGRN	N	**50.85 ± 0.86**	64.13 ± 0.70

Table 10 depicts the performance comparison between our model with different models on the *mini-ImageNet* dataset. The results lead to the following observations: This experiment is mainly divided into two parts. In the *5-way 1-shot* experiment, this model shows superior performance over other comparison models. In the *5-way 5-shot* experiment, this model is better than some models, but there is still a disparity between this model and a few models. This outcome can be attributed to the fact that the sample of the *mini-ImageNet* image dataset is relatively diverse and the feature relation is intricate, leading to the complexity in the calculation of the relation score and low accuracy of the final classification.

Table 11 shows the performance comparison between our model with different models on the *tiered-ImageNet* dataset. The following observations can be made according to the results:

(1) The accuracy of our model reaches 55.07% in the experiment of *5-way 1-shot* setting. This model is at least 0.8% higher than the other models, proving that

Table 11 Accuracy comparison between our model with different models on tiered-ImageNet (%). The best performing model is highlighted

Model	Fine	*5-way 1-shot*	*5-way 5-shot*
ECMT [20]	Y	48.19 ± 0.43	65.50 ± 0.39
Prototype net [25]	Y	48.58 ± 0.87	69.57 ± 0.75
BOIL [16]	N	49.35 ± 0.26	69.37 ± 0.12
MAML [4]	Y	51.67 ± 1.81	70.30 ± 1.75
Relation net [27]	Y	54.26 ± 0.27	**71.34 ± 0.28**
CGRN	N	**55.07 ± 0.20**	**71.34 ± 0.30**

our model promotes the model classification ability when the number of samples is relatively small.

(2) CGRN achieves an accuracy of 71.34% in the *5-way 5-shot* setting. It remains at the same level as these models, but it still has advantages in the face of some single sample distance calculation models.

5 Conclusions and future work

This paper proposes a few-shot learning model via relation network based on coarse-grained granulation, which is used to complete the few-shot classification task by representing the coarse-grained granulation and measuring the similarity of relation score. The model adopts the coarse-grained granulation strategy, which successfully mitigates the disadvantages of the traditional classification model with massive computation amounts and time complexity. Simultaneously, the model adopts the relation score as the measurement unit, thus overcoming the shortcomings of the traditional distance measurement model. A large number of comparative experiments on the *Omniglot*, *mini-ImageNet*, and *tiered-ImageNet* datasets show that this model can complete the classification task of target samples efficiently and quickly by calculating the relation score of coarse grain. Measurements between samples are crucial for the classification task. In the future, different similarity measures will be constructed by exploring different relations between samples to build efficient few-shot learning models.

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 Nos. 2021J011003 and 2021J011006.

References

1. Balte A, Pise N, Kulkarni P (2014) Meta-learning with landmarking: A survey. International Journal of Computer Applications 975:8887
2. Dietterich TG (1997) Machine-learning research. AI Mag 18(4):97–97
3. Edwards H, Storkey A (2017) Towards a neural statistician. Stat Sci 1050:20
4. 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
5. Goodfellow I, Bengio Y, Courville A (2016) Machine learning basics. Deep Learn 1:98–164
6. Haralick RM, Shanmugam K, Dinstein IH (1973) Textural features for image classification. IEEE Trans Syst Man Cybern 6(3):610–621
7. Kaiser Ł, Nachum O, Roy A, Bengio S (2017) Learning to remember rare events. arXiv:1703.03129
8. Khodizadeh-Nahari M, Ghadiri N, Baraani-Dastjerdi A, Sack JR (2021) A novel similarity measure for spatial entity resolution based on data granularity model: Managing inconsistencies in place descriptions. Appl Intell 51:6104–6123
9. Kingma DP, Ba J (2015) Adam: a method for stochastic optimization. In: International conference on learning representations
10. Koch G, Zemel R, Salakhutdinov R (2015) Siamese neural networks for one-shot image recognition. In: International conference on machine learning
11. Lake B, Salakhutdinov R, Gross J, Tenenbaum J (2011) One shot learning of simple visual concepts. In: Annual meeting of the cognitive science society
12. LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521(7553):436–444
13. Lu D, Weng Q (2007) A survey of image classification methods and techniques for improving classification performance. International Journal of Remote Sensing 28(5):823–870
14. Min S, Lee B, Yoon S (2017) Deep learning in bioinformatics. Brief Bioinform 18(5):851–869
15. Munkhdalai T, Yu H (2017) Meta networks. In: International conference on machine learning, pp 2554–2563
16. Oh J, Yoo H, Kim C, Yun SY (2021) Boil: Towards representation change for few-shot learning. In: International conference on learning representations
17. 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
18. Peng Y, Flach PA, Soares C, Brazdil P (2002) Improved dataset characterisation for meta-learning. In: International conference on discovery science, Springer, pp 141–152
19. Ravi S, Larochelle H (2016) Optimization as a model for few-shot learning. In: International conference on learning representations
20. Ravichandran A, Bhotika R, Soatto S (2019) Few-shot learning with embedded class models and shot-free meta training. In: IEEE/CVF International conference on computer vision, pp 331–339
21. Ren M, Triantafillou E, Ravi S, Snell J, Swersky K, Tenenbaum JB, Larochelle H, Zemel RS (2018) Meta-learning for semi-supervised few-shot classification. In: International conference on learning representations
22. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M et al (2015) Imagenet large scale visual recognition challenge. Int J Comput Vis 115(3):211–252
23. Santoro A, Bartunov S, Botvinick M, Wierstra D, Lillicrap T (2016) One-shot learning with memory-augmented neural networks. arXiv:1605.06065
24. Satorras VG, Estrach JB (2018) Few-shot learning with graph neural networks. In: International conference on learning representations
25. Snell J, Swersky K, Zemel R (2017) Prototypical networks for few-shot learning. In: International conference on neural information processing systems, pp 4080–4090
26. Snell J, Zemel R (2020) Bayesian few-shot classification with one-vs-each pólya-gamma augmented gaussian processes. In: International conference on learning representations
27. 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
28. Vilalta R, Drissi Y (2002) A perspective view and survey of meta-learning. Artif Intell Rev 18(2):77–95
29. Vinyals O, Blundell C, Lillicrap T, Wierstra D et al (2016) Matching networks for one shot learning. In: International

conference on neural information processing systems, pp 3630–3638
30. Wang Y, Yao Q, Kwok JT, Ni LM (2020) Generalizing from a few examples: a survey on few-shot learning. ACM Comput Surv 53(3):1–34
31. Wertheimer D, Hariharan B (2019) Few-shot learning with localization in realistic settings. In: IEEE/CVF Conference on computer vision and pattern recognition, pp 6558–6567
32. Xue Y, Deng Y (2021) Decision making under measure-based granular uncertainty with intuitionistic fuzzy sets. Appl Intell 51:6224–6233
33. Zhou A, Li Y (2021) Structural attention network for graph. Appl Intell 51:6255–6264
34. Zhou W, Liu C, Lei J, Yu L, Luo T (2021) HFNEt: Hierarchical feedback network with multilevel atrous spatial pyramid pooling for RGB-d saliency detection. Neurocomputing
35. Zhou W, Liu J, Lei J, Yu L, Hwang JN (2021) GMNEt: Graded-feature multilabel-learning network for RGB-thermal urban scene semantic segmentation. IEEE Trans Image Process 30:7790–7802

Xiao Jia is a master of Fujian Provincial Key Laboratory of Data Science and Intelligence Application, Minnan Normal University. His research interests include machine learning, deep learning and few-shot learning.

Yuling Su is currently a M.E. candidate with the School of Computer Science, Minnan Normal University, Fujian, China. Her current research interests include few-shot learning and hierarchical classification application in deep learning.

Hong Zhao received the Ph.D degree from Tianjin University, Tianjin, China, in 2019. She received her M.S. degree from Liaoning Normal University, Dalian, China, in 2006. She is currently a Professor of the School of Computer Science and the Key Laboratory of Data Science and Intelligence Application, Minnan Normal University, Zhangzhou, China. She has authored over 50 journal and conference papers in the areas of granular computing based machine learning and cost-sensitive learning. Her current research interests include rough sets, granular computing, and data mining for hierarchical classification.