Relative Position and Map Networks in Few-shot Learning for Image Classification

Zhiyu Xue, Zhenshan Xie, Zheng Xing, Lixin Duan
University of Electronic Science and Technology of China
No.2006, Xiyuan Ave, West Hi-Tech Zone
xzy990228@gmail.com, ivanxie1022@gmail.com, 3024979463@qq.com, lxduan@gmail.com

Abstract

Few-shot learning is an important research topic in image classification, which aims to train robust classifiers to categorize images coming from new classes where only a few labeled samples are available. Recently, metric learning based methods have achieved promising performance, and in those methods a distance metric is learned to directly compare query images against training samples. In this work, we consider finer information from image feature maps and propose a new approach. Specifically, we newly develop Relative Position Network (RPN) based on the attention mechanism to compare different pairs of activation cells from each query and training images, which captures their intrinsic correspondences. Moreover, we introduce Relative Map Network (RMN) to learn a distance metric based on the attention maps obtained from RPN, which better measures the similarity between query and training images. Extensive experiments demonstrate the effectiveness of our proposed method. Our codes will be released at `https://github.com/chrisyxue/RMN-RPN-for-FSL`.

1. Introduction

Till now, deep neural networks have achieved state-of-the-art performance on visual recognition tasks like image classification, object detection and semantic segmentation [4]. And the success of robust deep neural networks [11] mostly depend on abundant labeled instances with diverse visual variations. However, in practical applications, real-world problems with insufficient volume of training data has considerably impacted the performance of deep neural networks in a negative way. To deal with that, few-shot learning has been proposed to identify new classes from a few training samples and images, and significant progress has been made [21, 19] in the literature. However, the overfitting problem still remains challenging, due to lack of training data in new classes. Although the techniques of data augmentation and regularisation can alleviate the overfitting problem, they cannot fully solve it.

Previous work has been proposed to learn a transferable deep metric for comparing the relationship between support samples and query samples [19]. However, it is doubted that whether the manual metric function is the best way to measure the similarity between any two samples. Relation networks [20] use a CNN block to learn a similarity score by simply inputting the concatenation of two samples regardless of the difference of positions and maps. Motivated by [20], in this work, we come up with a simple framework that learns the difference between maps and the importance of positions. Specifically in proposed framework, we develop two modules: Relative Position Network (RPN) and Relative Map Network (RMN). RPN compares different pairs of activation cells from each support and query images based on the attention mechanism, which better captures their intrinsic correspondences. And RMN learns a distance metric based on the attention maps obtained from RPN in order to measure the similarity between images. Our contributions can be summarized as follows:

1. We propose a new framework for few-shot learning, which is based on metric learning and the attention mechanism.

2. Two modules, Relative Position Network (RPN) and Relative Map Network (RMN), are developed to better capture intrinsic correspondences between images, as well as to better measure the image similarity.

3. We conduct extensive experiments on the benchmark datasets, and promising results clearly show the effectiveness of our proposed framework.

2. Related Work

Few-shot Learning is to learn the concept from limited examples, and require an effective representation learning

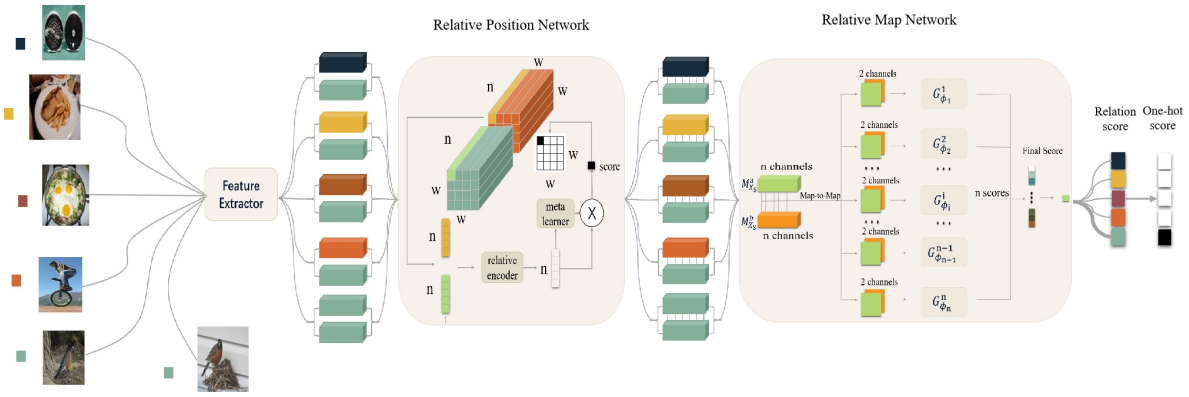

Figure 1. The overall architecture of our proposed framework. There are three parts of our method: a feature extractor, Relative Position Network (RPN), and Relative Map Network (RMN). The feature extractor extracts feature maps from images. RPN generates different weights to each position of feature maps during comparing process, based on attention mechanism. RMN is a new relation module in application for more efficient concatenation of feature maps.

with good ability to generalize information. For all existing methods [22], they can be divided into two categories: metric-based approaches [21] and gradient-based approaches [1]. The metric-based approaches are more related to our work which aims more at learning representations that can minimize the intra-class distances while maximizing the distance between different classes during few-shot learning [24]. In our work, we focus more on the relation network [20] and use the new method to learn the metric to help the model learn better from few samples.

Metric Learning plays a very crucial role in many visual tasks, as the performance of the deep learning models relies heavily on choosing a good metric. In few-shot learning, the the prior metric approaches [16, 21, 19] often result in complexity that metric need to be changed manually until the performance reached the ideal extent. In our work, instead of trying all the metrics to reach the state-of-the-art performance, we applied approaches based on meta-learning to let the model learn the best metric automatically.

Attention Mechanism is quite popular in many areas, such as image caption, voice recognition and machine translating [25, 18, 14]The attention mechanism prove to be useful in many computer vision related tasks. However, most approaches [5, 23] based on attention will only focus on the attention in individual images. In our work, we use the attention from different images to help compare the difference which will help the models learn important information.

3. Methodology

3.1. Problem Definition

For the task of few-shot classifier learning, the given datasets contain support set and query set. The support set contains C different image classes and K labeled samples per class, and few-shot learning aims to classify each unlabeled sample in query set according to the support set. This setting is called C-way K-shot classification. Following the setting of [21], we use two datasets meta training set D_{tr} and meta testing set D_{te}.($D_{tr} \cap D_{te} = \emptyset$). In this strategy of training, an episodic training paradigm is used to minimize the generalization error of D_{tr}. We divided the episodic in two steps, For N-way, sample N classes in meta-training set D_{tr} randomly,as classes in D_{tr} is equal to N. For K-shot, samples x_i in C randomly, which serve as support set $S = \{(x_i, y_i)\}_{i=1}^{m}(m = K \times |C|)$, and query set $Q = \{(x_i, y_i)\}_{i=1}^{n}(n = B \times |C|)$ is collected randomly for each class in C. ($Q \cap S = \emptyset$).

In this work, we adopt support set S as the measure criterion and use the query set Q to optimize the parameters of the model. The support set S and query set Q can also be abstracted in testing set D by taking the same measure proposed above to evaluate the performance. The training strategy is applied in our few-shot experiment(Section 4), and we also consider the settings of one-shot($K = 1$) and five-shot($K = 5$).

3.2. Relative Position Network

In Relative Position Network, considering different positions of a image may have different representative information, we think each position of feature maps must be treated differently during comparing process. Therefore, we come up with a new structure named Relative Position Network (RPN).

By generating different weights to the positions in feature maps, the Relative Position Network based on attention mechanism can determine which positions is important for model to compare. The structure of RPN is shown in Figure 1 as above. Samples x^S and x^Q are collected from the support set S from query set Q, respectively. And M_{x^S} and M_{x^Q} denote the feature maps of them. In the

support set, the position vector $v_{i,j}^S \in \mathbb{R}^{n\times 1}$ is the vector which locates in the i row and j column in feature maps $M_{x^S} \in \mathbb{R}^{b\times n\times w\times h}$, where b,n,w,h represent the batch size, channels, wideness and height of the feature maps. Similarly, we can define the position vector in query set by the same rules. To map the concatenation of two position vector $[v_{i,j}^S, v_{i,j}^Q]$ into a relative position vector and earn the intra-vector relationships of the relative position vector, We formalize as follows:

$$V_{i,j}^{s,q} = H([v_{i,j}^S, v_{i,j}^Q]) \tag{1}$$

$$w = W_2 \cdot \sigma(W_1 \cdot V_{i,j}^{s,q}) \tag{2}$$

where $H()$ is an encoder that can map the concatenation of two position vector $\{v_{i,j}^S, v_{i,j}^Q\}$ into a relative position vector. And $V_{i,j}^{s,q} \in \mathbb{R}^n$, $W_1 \in \mathbb{R}^{\frac{n}{r}\times n}$ and $W_2 \in \mathbb{R}^{\frac{n}{r}\times n}$ are the parameters of the meta learner, r is a scale ratio we need to fix in our experiments, and σ denotes a ReLU function Note that $\frac{n}{r}$ must be a integer.

$$Att_{i,j} = w^T V_{i,j}^{s,q} \tag{3}$$

$$M_{x^Q} := M_{x^Q} + Att \otimes M_{x^Q} \tag{4}$$

$\otimes$ indicates element-wise production operation and $Att_{i,j}$ reprerented a relative position score. And Eq.3 and Eq 4 take the attention operating only for query set, as the support set takes the role as criterion during comparing process. Also, a controlled set of experiments had done to demonstrate this assumption.

3.3. Relative Map Network

The original relation network simply concatenates the feature maps from support set and query set regardless of embodying the comparison principle adequately. Inspired by the structure of [7], we proposed a new relation module named Relative Map Network (RMN) to slove this problem. The structure can be shown in Figure 1. Our goal is to enable the network to compare these images individually and independently, since each single map in feature maps is different. So for the first step, the module selects two single maps from feature maps $M_{x_S}^i$ and $M_{x_Q}^i$ separately, where $i \in \{1, 2, 3...n\}$ represents the i-th channel of the feature maps. Then the embedding models $\hat{G} = \{G_{\phi_1}^1, G_{\phi_2}^2 ... G_{\phi_n}^n\}$ are trained to learn the parameters ϕ_i in the process. Each embedding model $G_{\phi_i}^i$ correspond inputting feature maps $M_{x_S}^i$ and $M_{x_Q}^i$ to learn a distance $p_i$between these feature maps, instead of designing a distance metric measure manually [9, 21]. Moreover, to compare the essential feature maps, a single full-connected layer is designed to compute a weighted sum $P_{S,Q}$ of every single output $G_{\phi_i}^i(M_{x_S}^i, M_{x_Q}^i)$, which is considered as the final similarity score between M_{x_S} and M_{x_Q}. This process can be represented as in Eq 5:

$$P_{S,Q} = Sig(\sum_{i=1}^{n} w_i G_{\phi_i}^i(M_{x_S}^i, M_{x_Q}^i)) \tag{5}$$

w_i denotes the weight needed to learn, and Sig presents the sigmoid function that can map the final score into the numerical range between 0 and 1.

In episodic training, following relation network [20], we use the mean square error (MSE) loss, which can be represented as follows:

$$MSE = \sum_{(x_S, y_S)\in S} \sum_{(x_Q, y_Q)\in Q} (P_{S,Q} - 1(y_S == y_Q))^2 \tag{6}$$

y_S and y_Q denote the targets of x_S and x_Q.

Table 1. Mean accuracies (%) of different methods on the MiniImageNet dataset. Results are obtained over 600 test episodes with 95% confidence intervals.

Model	MiniImageNet (5-way)	
	1-shot	5-shot
MATCHING NETS [21]	43.56±0.84	55.31±0.73
META LSTM [15]	43.44±0.77	60.60±0.71
MAML [3]	48.70±1.84	63.11±0.92
PROTOTYPICAL NETS [19]	49.42±0.78	68.20±0.66
META SGD [12]	50.47±1.87	64.03±0.94
RN [20]	50.44±0.82	65.32±0.70
GNN [17]	50.33±0.36	66.41±0.63
PABN [6]	51.87	65.37
TPN [13]	52.78±0.27	66.59±0.28
EGNN(No Trans) [8]	-	66.85
R2-D2 [2]	51.80±0.20	68.4±0.20
Ours	**53.35± 0.77**	**69.35± 0.61**

4. Experiments

4.1. Datasets

We evaluate our proposed approach with existing state-of-the-art baselines on the benchmark datasets (i.e., CIFAR-100 and miniImageNet).

Mini-Imagenet [21]is a dataset containing 60,000 colorful images coming from 100 classes, with 600 images in each class. In our experiments, we resize each image to a size of 84×84. Moreover, we use the same splits of [19], who employ 64 classes for meta-training, 16 for meta-validation and 20 for meta-testing.

CIFAR-FS [2] is randomly sampled from CIFAR-100 [10] by applying the same criteria as miniImagenet. The input size we use is 32×32, which is smaller than miniImagenet.

4.2. Implementation Details

Data augmentation: In our experiments, we use the random group of random resize crop, random vertical flip, ran-

Table 2. Mean accuracies (%) of different methods on the CIFAR-FS dataset. Results are obtained over 600 test episodes with 95% confidence intervals.

Model	CIFAR-FS (5-way)	
	1-shot	5-shot
MAML [3]	58.9±1.9	71.5±1.0
PROTOTYPICAL NETS [19]	55.5±0.7	72.0±0.6
RN [20]	55.0±1.0	69.3±0.8
GNN [17]	61.9	75.3
R2-D2 [2]	62.3±0.2	77.4±0.2
Ours	**61.43**	**76.16**

dom horizontal flip, and color jittering to achieve data augmentation. And we only apply data augmentation to query samples in the training set, as that support set is the criterion in metric learning which is better to keep stable.

Feature extraction: Our feature extractor contains four blocks. The first two blocks are the same as the blocks in relation network [20], which contain a convolution layer, a batch norm layer, a ReLU function and a max pooling layer which can change the size of feature maps into half. In the last two blocks, we use blocks as same as the blocks in ResNet [4].

RMN and RPN: In RMN, we use the combination of a convolution layer with 3×3 kernel without padding, a batch norm layer and a ReLU function, and two hidden layers for full connection layers. In RPN, we set the scale ratio for meta learner as $\frac{1}{2}$. Note that We initialize all networks randomly without involving additional datasets.

Optimization: As an optimizer, Adam is used as the same as [20]. The learning rate is initially set to 0.001 and later reduces to 0.5 times if the average accuracy over 300 validation episodes does not increase. The model is trained in a procedure with 5000 meta-training episodes, 300 meta-validation episodes and 600 meta-testing episodes. The total number of meta-training episodes is set as 500000.

4.3. Results and Analysis

Comparisons: We present the results of different methods on the MiniImageNet and CIFAR-FS datasets in Tables 1 and 2). We observe that our method outperforms other competitors by a noticeable margin on MiniImageNet, which clearly demonstrates the effectiveness of our method. However, although our method performs much better than GNN [17] on MiniImageNet, their results on CIFAR-FS are just comparable, possibly due to the dataset difference.

Ablation study and Visualization: To prove that our individual modules RPN and RMN can truly work, we take the ablation study in our framework. The results of this experiment show that both RPN and PMN can enhance the performance of relation network.

In the evaluation of RPN, we use the combination of a feature extractor, a relative position network and a relation module which is same as RN [20]. As for RMN, we use the same feature extractor and RMN.

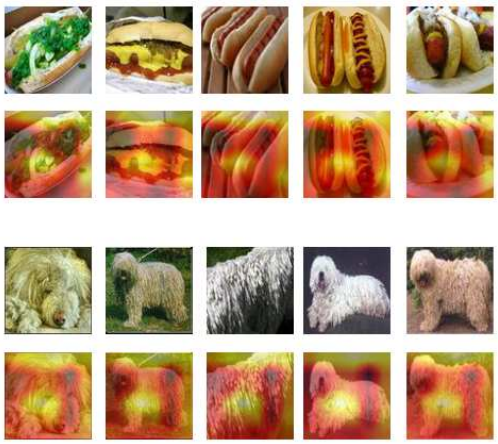

Figure 2. Visualization results of RPN in Mini-ImageNet.

All of the results can be shown in Table 3. Note that the numbers of RN w.r.t. $K = 1$ and 5 are directly from the original paper of RN [20], while the results w.r.t. $K = 3, 7$ and 10 are produced ourselves by using the publicly available open source code.

Table 3. Ablation study w.r.t. average accuracies (%) over 600 test episodes with 95% confidence intervals MiniImageNet in task 5-way K-shot about ablation study, where $K = 1, 3, 5, 7$ and 10.

Ave Acc	5-1	5-3	5-5	5-7	5-10
RN [20]	50.44	60.63	65.32	67.73	69.81
RPN	52.43	62.96	67.03	69.51	72.01
RMN	50.54	63.12	68.28	70.49	72.12
Ours	**53.35**	**63.94**	**69.35**	**70.87**	**73.17**

Furthermore, we show some visualization results in Fig 2 as an evidence to prove that our attention module RPN is workable. The samples from two categories are sampled from Mini-ImageNet.

5. Conclusion

In this paper, we propose a metric learning based method for few-shot learning. Unlike existing metric learning based work, we improve the learning of distance metrics by considering finer information of features maps of images through deep convolutional neural networks. Specifically, we develop a new module called Relative Position Network (RPN) based on the attention mechanism to more effectively compare different pairs of activation cells from the feature maps of query and supportimages. Moreover, we introduce Relative Map Network (RMN) to learn a distance metric based on those attention maps in order to better evaluate the similarity between images. Extensive experiments on benchmark datasets demonstrate the effectiveness of our proposed method over other state-of-the-art baselines.

References

[1] Samy Bengio, Yoshua Bengio, Jocelyn Cloutier, and Jan Gecsei. On the optimization of a synaptic learning rule. In *Preprints Conf. Optimality in Artificial and Biological Neural Networks*, pages 6–8. Univ. of Texas, 1992.

[2] Luca Bertinetto, Joao F Henriques, Philip HS Torr, and Andrea Vedaldi. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.

[3] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1126–1135. JMLR. org, 2017.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Sun Jian. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[5] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks.

[6] Huaxi Huang, Junjie Zhang, Jian Zhang, Qiang Wu, and Jingsong Xu. Compare more nuanced: Pairwise alignment bilinear network for few-shot fine-grained learning. *arXiv preprint arXiv:1904.03580*, 2019.

[7] Yunhun Jang, Hankook Lee, Sung Ju Hwang, and Jinwoo Shin. Learning what and where to transfer.

[8] Sungwoong Kim Chang D. Yoo Jongmin Kim, Taesup Kim. Edge-labeling graph neural network for few-shot learning. *arXiv preprint arXiv:1905.01436*, 2019.

[9] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, 2015.

[10] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

[11] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[12] Zhenguo Li, Fengwei Zhou, Chen Fei, and Li Hang. Meta-sgd: Learning to learn quickly for few shot learning.

[13] Yanbin Liu, Juho Lee, Minseop Park, Saehoon Kim, Eunho Yang, Sung Ju Hwang, and Yi Yang. Learning to propagate labels: Transductive propagation network for few-shot learning. *arXiv preprint arXiv:1805.10002*, 2018.

[14] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

[15] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. 2016.

[16] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pages 1842–1850, 2016.

[17] Victor Garcia Satorras and Joan Bruna Estrach. Few-shot learning with graph neural networks. 2018.

[18] Changhao Shan, Junbo Zhang, Yujun Wang, and Lei Xie. Attention-based end-to-end speech recognition on voice search. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4764–4768. IEEE, 2018.

[19] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017.

[20] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018.

[21] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Koray Kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning.

[22] Yaqing Wang and Quanming Yao. Few-shot learning: A survey. *arXiv preprint arXiv:1904.05046*, 2019.

[23] Sanghyun Woo, Jongchan Park, Joon Young Lee, and In So Kweon. Cbam: Convolutional block attention module.

[24] Chen Xing, Negar Rostamzadeh, Boris N Oreshkin, and Pedro O Pinheiro. Adaptive cross-modal few-shot learning. *arXiv preprint arXiv:1902.07104*, 2019.

[25] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.