Few-Shot Learning Based Image Classification

Saghar Irandoust, Abolfazl Eskandarpour, Mohammadhadi Mohandes, Sedigheh Razmpour

Introduction

Great success has been observed in visual recognition tasks such as image classification through deep learning methods. Using large numbers of labeled data in such supervised learning methods, limits their ability to recognize new classes or rare categories where numerous tested images may never exist. Few-shot learning aims to recognize new classes having seen only a few labeled examples [1]. There are many different approaches towards this task, and the one we are using is meta-learning [2]. In this framework, instead of learning how different classes look like, we learn *how to discriminate* between any group of given classes, called the *support set*. Our baseline method learns how to extract metrics from the *support set* and use it for relation comparison between the support set and a given test image [3].

Contribution

Problem :

- Data scarcity and annotation cost in some image classification problems with neural networks.

Solution:

- Using meta-learning, it *learns to learn* a metric to compare a few images within episodes, each designed to simulate the Few-shot setting.

- We show that using transfer learning in this context can help improve the accuracy of the baseline, as very strong feature extractors like ResNet [4] work much better than CNNs trained with very few data in the setting of Few-shot learning problem.

Proposed System structure

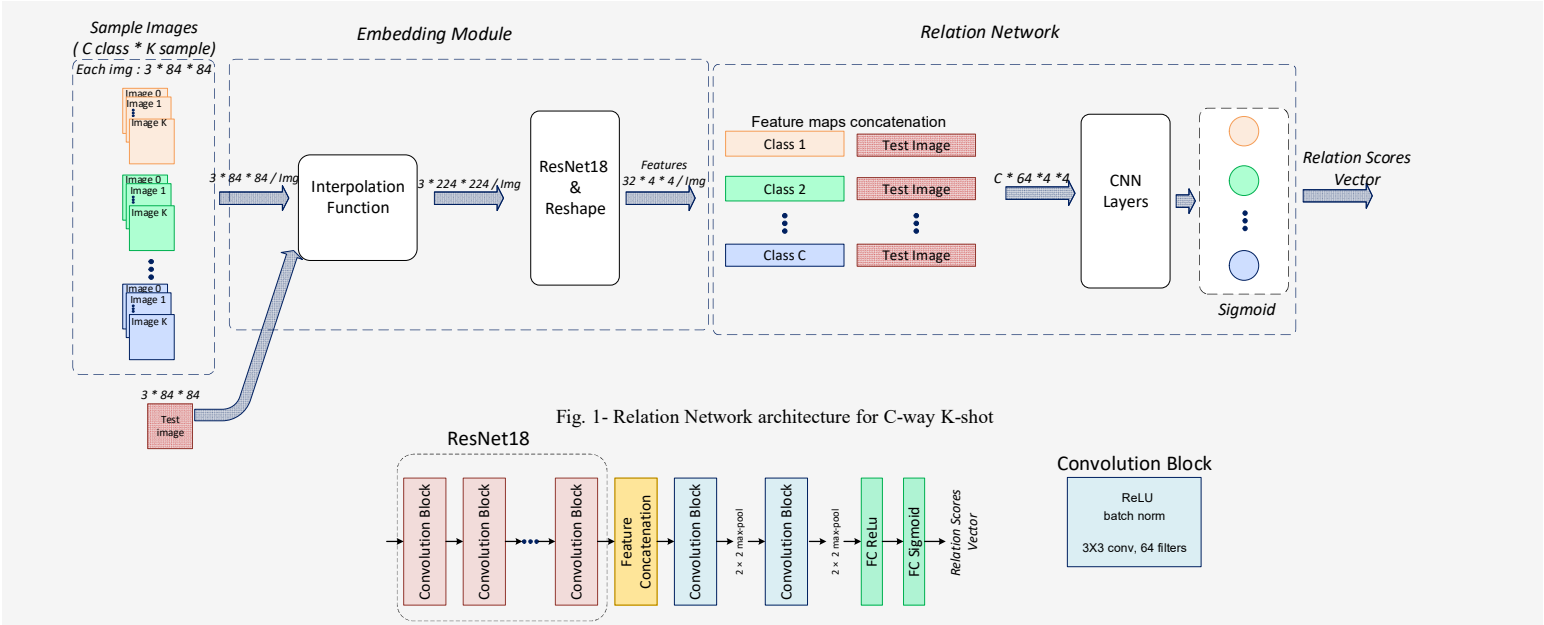

Fig. 1- Relation Network architecture for C-way K-shot

Fig.2 Relation Network Architecture Layers

Proposed Network Structure

Data: We consider the task of *few-shot classifier learning*. We have two datasets: a training dataset and a test dataset. The training and the test dataset have no overlap in their label spaces. In both training and test phases, data is represented in 2 sets. The *support set* and the *query set*. The support set contains K labeled examples for each of C unique classes (*C-way K-shot)*. The goal in the test phase is to recognize the labels of the query set using information from the support set..

Episodes: An effective way for exploiting training set in few-shot learning is episode based learning. In the training phase, in each *episode* we select C classes with K samples from each class randomly to make our support set. We select our query set from the remaining data in the same C classes. Therefore, each episode mimics the Few-shot scenario. In each training episode we update our model's parameters based on the evaluation of the model's performance on the training query set.

Networks: The system general architecture is illustrated in Fig.1. It is the combination of two parts, *embedding module* and *relation module.* The embedding module is used to extract features from both support and query sets. The feature map (f_φ) of each sample from the query and the mean feature map of each class from the support set are combined and fed into the relation module. The relation module ($g_\emptyset$), produces a scalar in range of 0 to 1 representing the similarity between x_i (i^{th} sample) and x_j (j^{th} class in query), which is called relation score.

$$r_{i,j} = g_\emptyset\left(C\left(f_\varphi(x_i), f_\varphi(x_j)\right)\right) \qquad (1)$$

Objective function: We use mean square error (MSE) loss (2) to train our model.

$$\varphi, \emptyset \leftarrow argmin_{\varphi,\emptyset} \sum_{i=1}^{m}\sum_{j=1}^{n}(r_{i,j} - 1(y_i == y_j))^2 \qquad (2)$$

Specifics: In our structure we finetune Resnet-18 removing its last FC layer, and use it as our embedding module. We interpolate input images in order to feed them into ResNet, and reshape the output feature vector into feature tensors.
The relation module is a 2 layer CNN network, using sigmoid as its last activation function. The detailed architecture is illustrated in Fig.2.

Result and Comparison

We implemented both the baseline paper [3] and our model. The baseline uses a 4 layer CNN encoder as its embedding module. The result of both under the same settings are mentioned in the table below. Dataset miniImageNet [4] is used in both systems' implementation.

Model	5-way Accuracy (Training episode = 1000, Test episode =20)	
	1-shot	5-shot
Baseline paper (encoder +RN)	36.10 %	40.10 %
Proposed structure (Resnet18 + RN)	**58.10 %**	**73.94 %**

Conclusion

We implemented a platform for few-shot learning, which exploits meta-learning for better accuracy in test set. We both implemented baseline [3] and also our proposed architecture. As it illustrated and compared in the table, the proposed architecture has better performance in some equal situation.
Resnet-18 is a powerful feature extractor which leads to better final feature concatenation and accuracy.

References

[1] Chen, Wei-Yu, et al. "A closer look at few-shot classification." International Conference on Learning Representations. 2019.

[2] Finn, Chelsea, et al. "Model-agnostic meta-learning for fast adaptation of deep networks." In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp. 1126-1135. JMLR. org, 2017.

[3] Sung, Flood, et al. "Learning to compare: Relation network for few-shot learning." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1199-1208. 2018.

[4] He, Kaiming,, et al. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.

[5] Vinyals, Oriol, et al. "Matching Networks for One Shot Learning." In Advances in Neural Information Processing Systems, pp. 3630-3638. 2016.