International Journal of Electronics and Electical Engineering

Volume 4 | Issue 1 Article 5

February 2022

A Modified Meta-Learner for Few-Shot Learning

Min Lu
School of Science, Shanghai Institute of Technology, Shanghai 201418, China, mlanran@163.com

Jingchao Yang
School of Electrical and Electronics Engineering, Shanghai Institute of Technology, Shanghai 201418, China, 1833385082@qq.com

Wenfeng Wang
Sino-Indian Joint research center of artificial intelligence and robotics, Interscience Institute of Management and Robotics, Bhubaneswar 752054, India;School of Electrical and Electronics Engineering, Shanghai Institute of Technology, Shanghai 201418, China, wangwenfeng@nimte.ac.cn

Bin Hu
Changsha Normal University, Changsha410111, China, hubin@csnu.edu.cn

Follow this and additional works at: https://www.interscience.in/ijeee

 Part of the Electrical and Electronics Commons

Recommended Citation

Lu, Min; Yang, Jingchao; Wang, Wenfeng; and Hu, Bin (2022) "A Modified Meta-Learner for Few-Shot Learning," *International Journal of Electronics and Electical Engineering*: Vol. 4: Iss. 1, Article 5.
DOI: 10.47893/IJEEE.2022.1184
Available at: https://www.interscience.in/ijeee/vol4/iss1/5

This Article is brought to you for free and open access by the Interscience Journals at Interscience Research Network. It has been accepted for inclusion in International Journal of Electronics and Electical Engineering by an authorized editor of Interscience Research Network. For more information, please contact sritampatnaik@gmail.com.

A Modified Meta-Learner for Few-Shot Learning

Min Lu[1], Jingchao Yang[2], Wenfeng Wang*[2,3] and Bin Hu[4]

[1]School of Science, Shanghai Institute of Technology, Shanghai 201418, China

[2]School of Electrical and Electronics Engineering, Shanghai Institute of Technology, Shanghai 201418, China

[3]Sino-Indian Joint research center of artificial intelligence and robotics, Interscience Institute of Management and Robotics, Bhubaneswar 752054, India

[4]Changsha Normal University, Changsha 410111, China

Abstract:

In this paper, we propose a modified meta-learning model, which is a LSTM-based meta-learning algorithm. Deep learning model learns through gradient back propagation. However, the gradient-based optimization method is not suitable for few-shot learning, and it is difficult to converge to an ideal state under fewer updates. Thus, a LSTM is used to learn a method to update the parameters of deep neural network (learner). On the basis of the original model, we modify the loss function L of the model considering the loss of the learner in the learning process, so that the performance of the model is more stable, the convergence is faster and the generalization ability is improved.

Keywords: deep neural network, modified meta-learning model, LSTM, Machine Learning

1 Introduction

Since its explosion in 2012, deep learning has revolutionized computer visionK. He et al.(2016), speech recognition A. v.d. Oord et al.(2016), translation Y. Wu et al.(2016), games V. Mnih et al.(2015) and Go D. Silver et al.(2016). However, as we all know, the success of deep learning is completely dependent on the amount of data and powerful computing resources. In the face of a new task, the model has to be retrained, which is very time-consuming and laborious. For example, in face recognition, an individual can often remember and recognize faces after looking only for few times L. A. Schmidt (2009), whereas today's deep learning requires thousands of images to do so. Therefore, how to enable artificial intelligence to have the ability of fast learning has become a frontier research problem.

Meta-learning, also known as learning to learn, aims to design models that can quickly learn new skills or adapt to new environments with a small number of training examples D. K. Naik and R. J. Mammone (1992),J. Schmidhuber(1987),S. Thrun and L. Pratt.(2012).There are three common approaches :1) Learn a valid distance metric:G. Koch et al.(2015), F.

Sung et al.(2018),O. Vinyalset al.(2016). (metric-based); 2) a circular network using external or internal memory: T. Munkhdalai and H. Yu.(2017),A. Santoroet al.(2016). (model based); 3) Optimize model parameters to achieve fast learning: C. Finn et al.(2017),A. Nicholet al.(2018),S. Ravi and H. Larochelle.(2016). (Optimization based). We expect a good meta-learning model to adapt or generalize well to new tasks and new environments that are never encountered during training. The adaptation process, which is essentially a small learning process, occurs during testing, but exposure to the new task configuration is limited. Eventually, the adjusted model can perform the new task.

Meta-Learning, or so-called Learning to learn, has become another important research branch in Machine Learning. Different from traditional deep learning, meta-learning can be used to solve one-to-many problems and has a better performance in few-shot learning which only few samples are available in each class. In these tasks, meta-learning is designed to quickly form a relatively reliable model through very limited samples. In this paper, we propose a modified LSTM-based meta-learning model, which can initialize and update the parameters of classifier (learner) considering both short-term knowledge of one task and long-term knowledge across multiple tasks. We reconstruct a Compound loss function to make up for the deficiency caused by the separate one in original model aiming for a quick start and better stability, without taking expensive operation. Our modification enables meta-learner to perform better under few-updates. Experiments conducted on the Mini-ImageNet demonstrate the improved accuracies.

2 Model

In this section, we will go through the details of modified meta-learner model.

2.1 Model description

Our model is a modification of Meta-Learner proposed by S. Ravi and H. Larochelle. (2016). Two neural networks are involved in this model. Let's denote the learner, which is the model for dealing the task, as M with the parameters θ, meta-learner, which updates learner's parameters, as R with the parameters Θ. Learner $M$is a CNN with 4 convolutional layers, each of which is a 3 × 3convolution with 32 filters, followed by batch normalization, a ReLU, and a2 × 2 max-pooling. Generally, a gradient-based update of learner can be expressed as

$$\theta_t = \theta_{t-1} - \alpha_t \nabla_{\theta_{t-1}} \mathcal{L}_t \quad (1)$$

Where $t$is the time step and α_t is the learning rate of current step, and let's take a close look at cell state update in LSTM:S. Hochreiter and J. Schmidhuber (1997).

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (2)$$

And with $f_t = 1$, $i_t = \alpha_t$, $c_t = \theta_t$, $\tilde{c}_t = -\nabla_{\theta_{t-1}} \mathcal{L}_t$we can have

$$c_t = \theta_t = \theta_{t-1} - \alpha_t \nabla_{\theta_{t-1}} \mathcal{L}_t \quad (3)$$

Which means, cell state in LSTM is the parameters of learner, and LSTM will update these parameters through updating its own cell state, so that LSTM

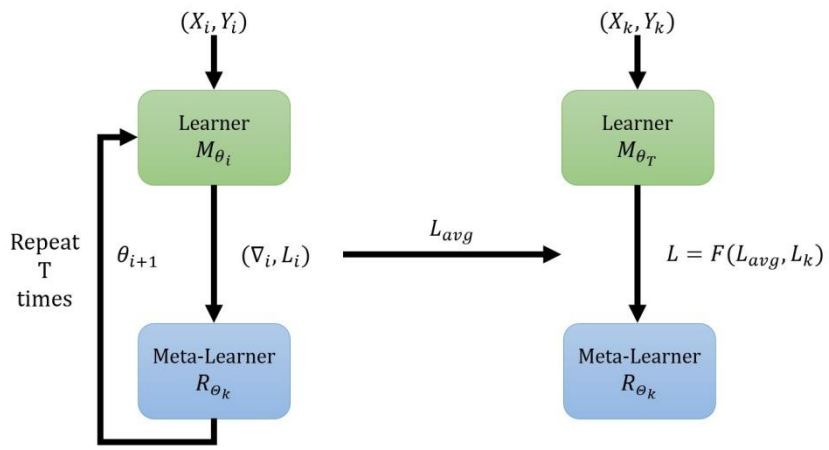

Figure 1: Graph of the Meta-learner's training process for one episode. (X_i, Y_i) and (X_k, Y_k) are the samples from the D_{train}, D_{test}respectively.

can value how a history of gradients benefits the gradient update and enable the learner to adapt to a new task easily. Thus, for meta-learner *R*we use a 2-layer LSTM, where the first layer is a normal LSTM and the second layer is meta-lstm. The gradient and loss are preprocessed and input into the first layer LSTM to get the regular gradient coordinates and the second layer of LSTM is used to implement the update of parameters θ. The meta-learner shares the same cross entropy loss as learner, and updated every episode by Adaptive momentum.

2.2 Training & Update

Now, let's explain the training process of the model where the modifications are made. The training process is shown as Figure1. In a single episode, learner will be trained on D_{train} for a series update. For each epoch, the loss function can be described as

$$\mathcal{L}_i = \mathcal{L}(M(X_i; \theta_{t-1}), Y_i) \quad (4)$$

Then we can have the average of the loss over a training episode

$$\mathcal{L}_{avg} = \frac{1}{T}\sum_{i=1}^{T} \mathcal{L}(M(X_i; \theta_{t-1}), Y_i) \quad (5)$$

The final state of the learner parameter $\theta_T$is used to train the meta-learner on D_{test} and we can have the loss of meta-learner

$$\mathcal{L}_k = \mathcal{L}(M(X_k; \theta_T), Y_k) \quad (6)$$

In the original model, only $\mathcal{L}_k$is considered to update the parameters of meta-learner, which means, the parameters of Meta-learner are only updated at the end of an episode, the purpose of asynchronous update was to prevent model instability caused by frequent parameters change. But our observation is that the learner itself also exhibits instability during one episode, especially in the first few rounds. There is a large discretization in accuracy and loss of

Learner in training, so we think that the loss of Learner $\mathcal{L}_k$ cannotreflect the performance of Learner in an episode. In other words, it may not be very appropriate to update the meta-learner's parameters by using the separate loss $\mathcal{L}_k$. Thus, we take $\mathcal{L}_{avg}$ in to consideration and a compound loss function is constructed as

$$\mathcal{L}_T = W_1\mathcal{L}_k + W_2\mathcal{L}_{avg} \qquad (7)$$

where $W1$ and $W2$ are adjustable weights, here we let $W1 + W2 \approx 1$ so that training rate before and after modification is as consistent as possible. $\mathcal{L}_T$ is simply calculated as a weighted sum, but with this modification, we can assess the performance of the model more accurately and update the parameters more effectively over an episode.

3 Experiment & Results

In our experiment, Mini-Image net, which is proposed by O. Vinyalset al. (2016), is created according to S. Ravi and H. Larochelle. (2016) we select 64 classes with 600 samples in each class randomly for training, 15 and 20 classes for validation and testing respectively. We denote them as $\mathcal{D}_{mata-train}$, $\mathcal{D}_{meta-validation}$, and $\mathcal{D}_{meta-testing}$. Each meta-set $\mathcal{D}$ contains several regular datasets$D \in \mathcal{D}$.

The k-shot, N-class classification task is used examining the performance of model, where for each dataset D, the training set consists of k labelled examples for each of N classes, meaning that D_{train} consists of K*N examples. Here, we consider the 5-shot,5-class classification as shown in Figure2. First, we need to sample 5 classes from the meta-set and 5 samples from each of those classes to form D_{train}. Among the rest of samples in these classes, we select 15 of them per class to form D_{test}.

Figure 2: An example of 5-shot 5-class image classification task (Image thumbnails are from Image Net)

During experiment, we use seeds to randomly generate data sets, and the same seeds will generate the same data sets. Our model will be trained and update for fixed

times. At the same time, as a contrast, we train the model on two different algorithms (Original meta-learner and Matching network FCE) simultaneously using the same seed, which means they are trained under same conditions. After a certain number of updates, we test the performance of the models on the test set. In addition, the models are saved and fine-tuned to adapt to the new task, which means the different data set generated by different seed. Similarly, the performance of the models on the new task is evaluated for their generalization ability.

For both meta-learner and our modification version, a CNN and a modified LSTM are used as learner and meta-learner respectively as mentioned above. All the networks' Settings (including Matching Network) are referenced toS. Ravi and H. Larochelle. (2016). The result of our experiment is shown in Table1 and Table2.

Table 1: Average accuracies over 500 episodes (* means the model is saved and trained for another 500 episodes on different datasets)

Model	Accuracy	Accuracy(Fine-tune)*
Modified Meta-Learner	37.17+-6.9%	39.16+-7.7%
Meta-Learner	36.90+-7.6%	38.37+-8.8%
Matching Net FCE	37.02+-7.2%	-

Table 2: Average accuracies over 1000 episodes (* means the model is saved and trained for another 1000 episodes on different datasets)

Model	Accuracy	Accuracy(Fine-tune)*
Modified Meta-Learner	37.47+-7.9%	40.28+-7.7%
Meta-Learner	36.72+-7.6%	36.43+-7.2%
Matching Net FCE	37.26+-6.8%	-

4 Discussion

It can be seen from the experimental results that the modified meta-learner can converge more quickly with better stability, and performs better than the original one in transfer learning. This is consistent with what we expected. We notice that the matching network also has a good

performance in the test, which is also reflected in the works of S. Ravi and H. Larochelle. (2016). The optimization of the model by modifying the loss function is very limited. When the episodes of training are large enough, we think that the modified model and the original model should have the same performance. The results of accuracies on fine-tuned tasks show the effectiveness of the modification. However, our purpose of modifying the model is not to improve its accuracy in few-shot learning, but to improve the stability and fast learning ability. This is also the goal of meta-learning: to design models that can quickly learn new skills or adapt to new environments through a small number of training examples. Considering the effects of multiple losses as much as possible is beneficial to the algorithm to update the parameters of the model effectively, which is especially obvious at the beginning of training or transfer learning. With a quick start, we can save a lot of time in the training process, and the model can reach an ideal level faster. However, our modified model still has some shortcomings, that is, how to determine the weight of the compound loss function. In the model, the performance of the base learner in small sample learning is unpredictable. How to effectively identify the biased results and adjust the corresponding loss weights is a problem that should be carefully considered. This also leads to inappropriate weight settings that make the model even worse. We cannot find a more reliable way to solve this problem at present maybe we can learn some performance in the training process through another neural network and adjust these weights dynamically, but there is no doubt that this will bring more computational cost.

We believe that the key to improve the performance of the meta-learner is to modify the algorithm itself. The assumption of gradient independence in the original algorithm is to reduce the calculation, but it will inevitably bring some adverse effects. How to remove this gradient independence without a considerably expensive operation will be a breakthrough point. In addition, the loss function used in the meta-learner is still the loss function under the learner. How to construct a more suitable loss function for meta-learner is a valuable and challenging task.

5 Conclusion

We propose a modified meta-learner. based on the original one [10], we eliminate the instability caused by the individual loss function by using the compound loss function and accelerate the training of the model. Experiments show that our method outperforms the original model and other algorithms in the case of less training episodes and transfer learning. which means a quick start and better generalization ability. Our research is still based on few-shot learning, but the purpose of meta-learning is not limited to this. It will be our future task to modify the algorithm itself and construct a more suitable loss function to make the model perform well in various environments.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 770–778.

[2] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. 2016.*arXiv preprint arXiv:1609.03499*.

[3] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al. Google's neural machine translation system: Bridging the gap between human and machine translation, 2016.*arXiv preprint arXiv:1609.08144*.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 2015, 518(7540):529–533.

[5] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*,2016.529(7587):484–489.

[6] L. A. Schmidt. *Meaning and compositionality as statistical induction of categories and constraints*. PhD thesis, Massachusetts Institute of Technology, 2009.

[7] D. K. Naik and R. J. Mammone. Meta-neural networks that learn by learning. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, IEEE, 1992 volume 1, pages 437–442.

[8] J. Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, TechnischeUniversit¨atMu¨nchen, 1987.

[9] S. Thrun and L. Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

[10] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, 2015,volume 2. Lille.

[11] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*,2018, pages 1199–1208.

[12] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra. Matching networks for one shot learning, 2016.*arXiv preprint arXiv:1606.04080.*

[13] T. Munkhdalai and H. Yu. Meta networks. In *International Conference on Machine Learning*. PMLR, 2017, pages 2554–2563.

[14] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. Metalearning with memory-augmented neural networks. In *International conference on machine learning*,PMLR, 2016. pages 1842–1850.

[15] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*. PMLR, 2017, pages 1126–1135.

[16] A. Nichol, J. Achiam, and J. Schulman. On first-order meta-learning algorithms. 2018,*arXiv preprint arXiv:1803.02999*.

[17] S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. 2016.

[18] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 1997, 9(8):1735–1780.