Few Shot Learning Based on the Street View House Numbers (SVHN) Dataset

Rundong Yang[1], Yancong Deng[2](✉), Anqi Zhu[3], Xin Tong[4], and Zhihao Chen[5]

[1] Hebei University of Technology, Hebei, China
[2] University of California, San Diego, USA
yad002@eng.ucsd.edu
[3] University of New South Wales, Sydney, Australia
z5141541@unsw.edu.au
[4] University of Illinois at Urbana Champaign, Champaign, IL, USA
xintong2@illinois.edu
[5] Shanghai University, Shanghai, China

Abstract. In recent years, deep learning model has made remarkable achievements in image, voice, text recognition and other fields. However, deep learning model relies heavily on large number of labeled data, which limits its application in the special field of data shortage.

For the practical situation such as lack of data, many scholars carry out research on the few shot learning methods, and there are many typical research directions, among which model-agnostic meta-learning (MAML) is one of them. Aiming at the few shot learning method, this paper systematically expounds the current main research methods on few shot learning, the algorithm of MAML and implements the MAML on the SVHN dataset.

Keywords: Few shot learning · Meta learning · MAML · Digit detection

1 Introduction

Deep learning has made significant progress in areas such as computer vision, but this is based on the fact that they have a large amount of labeled data. However, it is impractical to obtain a large amount of data in real life, for example, in the fields of medicine and security, labeling data is scarce, and the cost of obtaining label data is also very large. For deep learning methods, fitting a more complex model requires more data. Generally speaking, in the case of small amount of data, the training effects of deep learning are not good, and the recognition performance of new kind of samples is also poor.

Since humans can learn from a small number of samples, we believe that deep learning technologies will also get rid of their dependence on the amount of data in the future. Therefore, few shot learning has become a very important research direction in the field of deep learning in recent years.

H. Jiang et al. (Eds.): ICECI 2020, LNICST 368, pp. 86–102, 2021.
https://doi.org/10.1007/978-3-030-73429-9_6

To solve such problems, many methods have been carried out. This paper introduces several current few shot learning methods based on deep neural network with few samples, such as prototype network-based method, optimization-based method, and migration-based learning method.

Among these methods, meta learning is the method with high performance and it was inspired by Human beings. Human beings can not only learn from a small number of samples, but also implement experiences on other kinds of samples. Like human beings, meta learning does not consider new tasks in isolation, but uses previous experience to quickly learn new tasks, so as to achieve the ability to learn how to learn, which can well solve the problem of task-specific categories in few shot learning. In this paper, the model-agnostic meta-learning (MAML) algorithm and the results of experiment on SVHN dataset are described in detail.

2 Related Work

Meta learning, also known as learning to learn, refers to the use of previous experience to quickly learn new tasks without considering new tasks in isolation.

In 2001, Memory based neural network is proved to be useful in meta learning (S. Hochreiter, 2001) [3]. The siamese neural network is composed of two identical convolution neural networks, and the similarity between the two images is calculated by comparing the loss function with the paired samples input(Gregory Koch, 2015) [4]. Prototype network can project each sample of each category into the same space. For each kind of sample, their center point is extracted as the prototype, and the distance between the sample to be classified and the prototype of each category is calculated by Euclidean distance and the distance can be used to classify(Jake Snell, 2017) [10].

A memory enhancement network based on long-term memory network (LSTM) was proposed in 2016 [9]. Memory enhancement network train data as a sequence, and the last label is also used as network input, and external storage is added to store the last input. This enables the labels to establish a relationship with the input when the next input is back propagation, so that the subsequent input can obtain the relevant image through external memory for comparison, so as to achieve better prediction(Adam Santoro, 2016). Other scholars used LSTM as a meta learner, and took the initial parameters, learning rate and loss gradient of the learner as the state of LSTM to learn the initialization parameters and parameter update rules of the learner(Ravi and Larochelle, 2017) [8].

Model agnostic meta learning (MAML) algorithm trains on a large number of different tasks, and can quickly adapt to new tasks through a small number of gradient steps. Compared with the previous meta learning method, this method does not introduce additional parameters and has no restrictions on the structure of the model. Only gradient is used to update the weight of the learner (Chelsea Finn, 2017) [1]. Based on the MAML and neglecting the quadratic differential, Alex Nichol proposed a meta learning method to find the initialization parameters of the neural network, Reptile. Reptile only needs to execute the random gradient descent algorithm on each task, and does not need to compute quadratic differential like the MAML, so it consumes less computation and memory(Alex Nichol, 2018) [6].

3 MAML Algorithm Introduction

MAML directly optimizes the initialization parameters of learners. After updating the parameters by using one or more gradient iterative steps calculated from a small amount of data from the new task, the learner has the maximum generalization performance on the new task, so it has the ability to learn how to learn. MAML can be easily combined with fully connected neural network, convolution neural network or recurrent neural network, and it can also be used in various loss functions, including differential supervised loss and non differential reinforcement learning objectives. For this reason, the MAML is called Model agnostic meta learning.

3.1 Convolution Neural Network

The learner used in this paper is a convolution neural network [5,13], which consists of four convolution layers and a full connection layer. The convolution kernel sizes are (3 * 3), (3 * 3), (3 * 3), (2 * 2), respectively. After each convolution layer, a RELU activation function layer and a batch normalization layer are applied. The batch normalization layer can prevent over fitting, improve the learning rate, and training speed [7]. The structure of convolution neural network used in this paper is shown in Fig. 1.

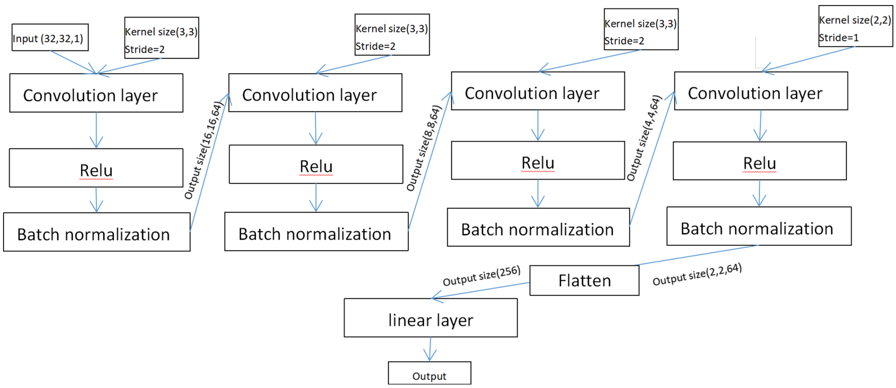


Fig. 1. Convolution neural network

3.2 Pseudo Code

The initial parameter optimization process after one gradient iteration step is shown in Fig. 2. Each task is represented by T_i, $p(T)$ as the distribution function of the task set, $f(\theta)$ as the few shot learner that maps the original pixel characteristic value x of the image to the output value, and θ as the parameter value of the learner. Extract task T_1 from $p(T)$. After a gradient iteration, a new temporary parameter θ_1' that adapts to the specific task is calculated. Then, new samples from task T_1 is taken for testing, and the loss $L_{T_1} f(\theta_1')$ of corresponding task T_1 is calculated. Then, the loss of different tasks will be kept generating until we get the loss of new task T_n. Finally, the sum of test loss in different tasks is used for the parameters optimization of the meta learning process. The parameters of the learner are updated by the gradient descent method, and finally a set of initialization parameters are obtained.

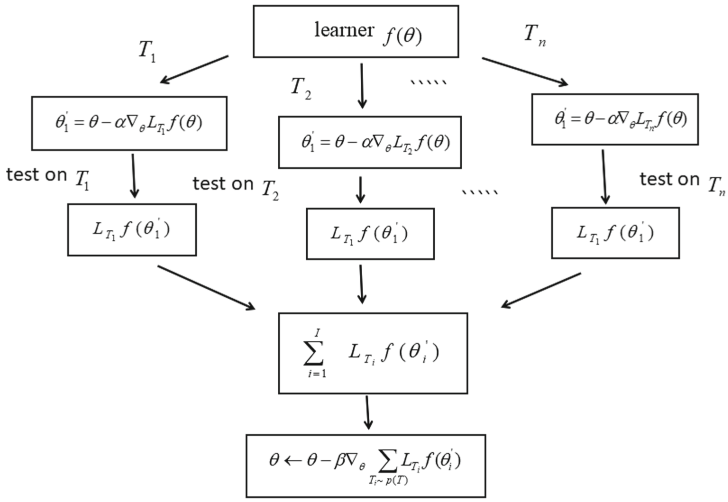


Fig. 2. Flowchart of MAML

To be specific, for the task of N way K shot, first determine the number of tasks in each step, then for each task, select N classes from the dataset, and select $K + Q$ images from each class as the support set and query set(Q is the number of pictures in query set). The support set is input into the convolution neural network, and the cross entropy formula is used to calculate the loss and update the temporary parameters.

$$L_{T_i}(f_\theta) = \sum_{x^{(j)}, y^{(j)} \sim T_j} y^{(j)} \log f_\theta\left(x^{(j)}\right) + \left(1 - y^{(j)}\right) \log\left(1 - f_\theta\left(x^{(j)}\right)\right)$$

Then the query set is input into the convolution neural network with temporary parameters, and the loss is calculated. After all the tasks are trained, the average value of all the losses obtained by the query set is calculated, and the value obtained is used to update the initialization parameters of the meta learner through the Adam optimizer.

To sum up, the above can be summarized as the following pseudo codes.

Algorithm 1. Model agnostic meta learning [1]

Require: α, β: learning rate
Require: I: the number of task of per batch
Require: $p(T)$: distribution of tasks
Require: ADAM(θ, L): using adam optimizer to update
Randomly initialize θ
while not done **do**
  sample batch of tasks $T_i \sim p(T)$
  for all T_i **do**
    $S_i = \left\{x^{(i)}, y^{(j)}\right\}$ # support set
    $f_\theta\left(S_i\right)$ # train model with parameters θ
    $L_{T_i}\left(f_\theta\right) = \sum_{x^{(j)}, y^{(j)}} y^{(j)} \log f_\theta\left(x^{(j)}\right) + (1 - y^{(j)}) \log(1 - f_\theta(x^{(j)}))$ # calculate cross-entropy
    $\theta_i' = \theta - \alpha \nabla_\theta L_{S_i}\left(f_\theta\right)$ # update parameters of the model temporarily
    $Q_i = \left\{x^{(i)}, y^{(j)}\right\}$ # query set
    $f_{\theta'}\left(Q_i\right)$ # train model with parameters θ
    $L_{T_i}\left(f_\theta\right) = \sum_{x^{(j)}, y^{(j)}} y^{(j)} \log f_\theta(x^{(j)}) + (1 - y^{(j)}) \log(1 - f_\theta(x^{(j)}))$ # calculate cross-entropy
  end for
  $L = \sum_{i=1}^{l} L_{T_i}\left(f_{\theta_i'}\right) / I$ # calculate the total loss of the batch
  ADAM(θ, L) # update parameters of the model
end while

3.3 Apply to SVHN

In this paper, we use Python to apply MAML to the SVHN dataset to explore how the MAML performs with the digit recognition in natural scene. Firstly, the convolution neural network introduced in Sect. 3.1 is built by using Pytorch. Then we write code to extract tasks from SVHN dataset, each task includes N categories, each category includes $K + Q$ pictures (K is the number of pictures in support set, Q is the number of pictures in query set). An iteration consists of 32 tasks. For each iteration step, we use the support set of each task to update the temporary parameters 5 times. After that, we use the query set to get the loss values, and the average value of the 32 loss values is used to update the parameters.

During the test, we extract one task and update the parameters 10 times using the support set and get the test accuracy using the query set. In addition, we use the Visdom service to visualize the experiment. The hyper parameters used in the program are shown in Table 1.

Table 1. Hyper parameters

Task number	Meta learning rate	Training learning rate	Update step	Test update step
32	0.001	0.4	5	10

4 Experiment

4.1 Preparation

SVHN (street view house number) dataset is the real world data, in order to develop the machine learning and target recognition algorithm [2]. Similar to MNIST, but larger. And they all come from problems that are obviously more difficult and unsolved in the real world. The dataset is from Google Street view pictures.

There are 10 kinds of SVHN. The number 1 is labeled as 1 and the number 9 as 9. There are 73257 training sets and 26032 test sets. There are also 531131 additional, simpler sampling data that can be used as training sets.

Fig. 3. SVHN dataset

Although there are only 10 categories of images in SVHN, the number of images in each category is large, and the selected images will be relabeled in each task. So it can meet the requirements of diverse data types in few shot learning experiments. In addition, the SVHN dataset is shot from the real world. Therefore, compared with the Omniglot dataset commonly used in the few shot learning experiment, few shot learning experiment of SVHN is more challenging. Therefore, in this paper, we use SVHN dataset to do experiments.

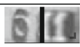
Next, we extract the corresponding data from SVHN to complete the few shot learning and one shot learning experiments.

4.2 Two Way One Shot Learning

In order to ensure the accuracy of the test accuracy, the training dataset and the test dataset should be inconsistent [12]. So we divide the ten categories of SVHN dataset into two parts, one includes eight categories of images for training, named training part, the other includes the other two categories of images for testing, named testing part.

For the training tasks of 2-way 1-shot, for each task, we will extract two kinds of images from the eight kinds of images in the training part, and then take one images from the two kinds of images as the support set and 15 images as the query set respectively [11]. For the testing tasks, the two categories images in testing part are all used. And the number of images in the testing tasks is the same as that in the training tasks.

Table 2. Examples of tasks (MAML, SVHN, 2-way 1-shot)

		Support set	Query set
Training Data	Task1		
	Task2		
	Task3		
	...	...	...
Testing Data	Task1		
	Task2		
	...	...	...

By inputting the above data into the MAML model, we record the training accuracy and loss of each query set in training step. As can be seen from Fig. 4, the growth rate is faster in steps 0–1300, and it starts to rise slowly after steps 1300. Finally, the accuracy of training steps is stable at 89.9% ± 5.2% after 3500 steps. It should be noted that the accuracy of training steps is not as reliable as that of testing steps, because there may be over fitting phenomenon due to the small number of datasets. The trend of loss is similar to that of accuracy (Fig. 5).

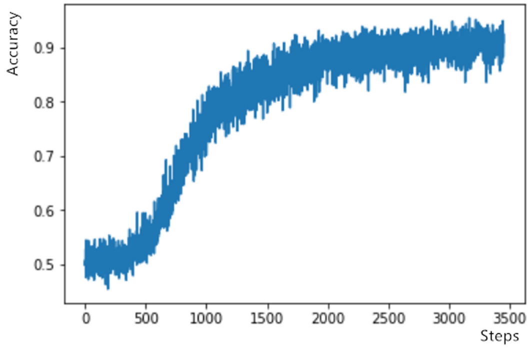


Fig. 4. Accuracy of training steps (MAML, SVHN, 2-way 1-shot)

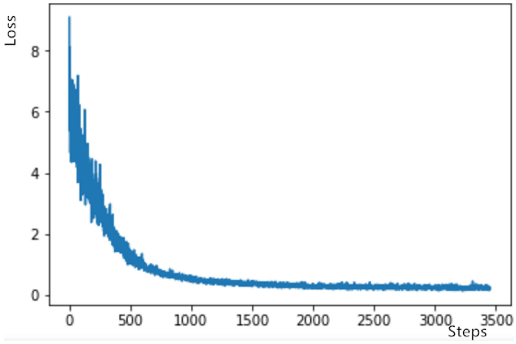


Fig. 5. Loss of training steps (MAML, SVHN, 2-way 1-shot)

Take the parameters obtained in the training steps as the initial parameters of the test step, and carry out one test step every 500 training steps. It can be seen from Table 3 that after 3000 iterations, the test accuracy of the MAML model in this paper can reach 70.3% for 2-way 1-shot. We extract the query set in test step, and compare the predicted value and the true label of the images (Fig. 6). Among the ten images, the predicted values of the first, the fourth and the ninth are not consistent with the true labels, with an accuracy of 70%, which is basically consistent with the test accuracy of the MAML model.

4.3 Two Way Five Shot Learning

Similar to the method of extracting data in 2-way 1-shot, we extract two kinds of images from SVHN dataset for each task. The difference is that five images are extracted from each type of images as the support set. Due to the larger amount of data in few shot learning, theoretically, the accuracy of the test will be higher [12].

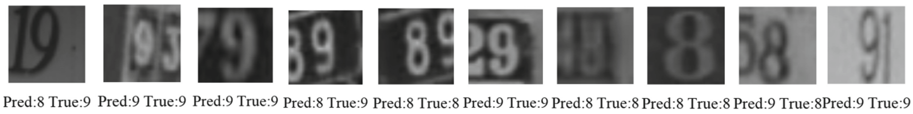


Fig. 6. Predictions and true labels (MAML, SVHN, 2-way 1-shot). The true label of the first picture and the fourth picture is 9, but the prediction label is 8. The true label of the ninth picture is 8, and the prediction label is 9. The true labels of other pictures are consistent with the prediction labels.

The experimental data of 2-way 5-shot is shown in the Fig. 7 and Fig. 8. It can be seen from the figures that the accuracy of 2-way 5-shot experiment increases and the loss decreases significantly faster than 2 way 1 shot experiment. Step 0–400 is the fastest range of accuracy and loss. After step 400, accuracy rises slowly and stabilizes at 94.3% ± 0.8%. Compared with Fig. 4 and Fig. 5, it can also be found that the training accuracy and loss of the 2-way 1-shot experiment have a larger fluctuation range.

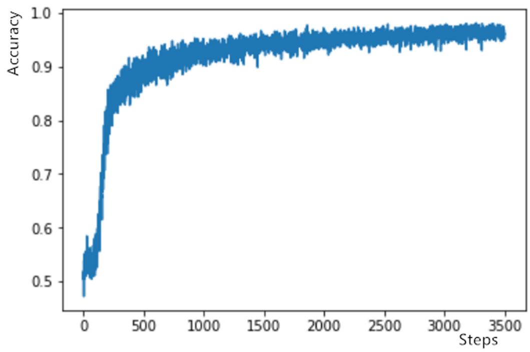


Fig. 7. Accuracy of training steps (MAML, SVHN, 2-way 5-shot)

The results of the test steps are shown in Table 3. After 1500 iterations, the accuracy of the test is 80.1% ± 0.4%. Figure 9 shows ten images extracted from the query set of the test task. From the images, the predicted values of the fifth and seventh images are different from the real labels, and other predicted values are consistent with the real labels, with an accuracy of 80%, consistent with the test results.

4.4 Five Way Five Shot Learning

For the 5-way-5 shot experiment, we have five kinds of images in the training part and the test part of the SVHN dataset respectively. The other algorithms are the same as the two classification experiments. The final experimental results are shown in the figures below.

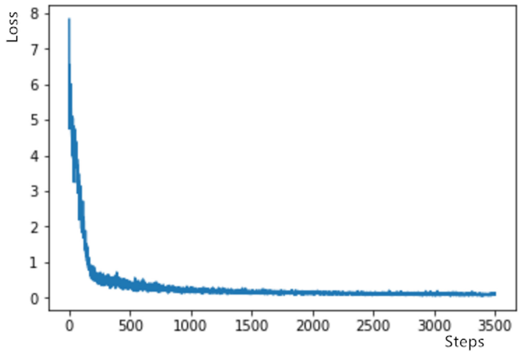


Fig. 8. Loss of training steps (MAML, SVHN, 2-way 5-shot)

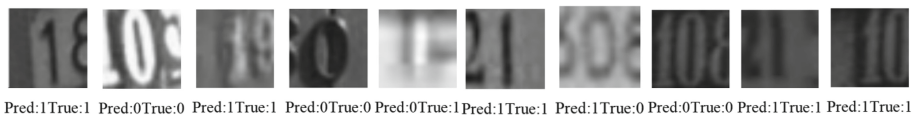


Fig. 9. Predictions and true labels(MAML, SVHN, 2-way 5-shot). There are two false prediction, such as the true label of the fifth picture is 1, but the prediction label is 0. The true label of the seventh picture is 0, and the prediction label is 1. The true labels of other pictures are consistent with the prediction labels.

Based on these figures, the test results can be summarized as follows: after 3000 iterations the training accuracy can be stabilized at $98\% \pm 0.5\%$, the accuracy of the test is $53.6\% \pm 3\%$.

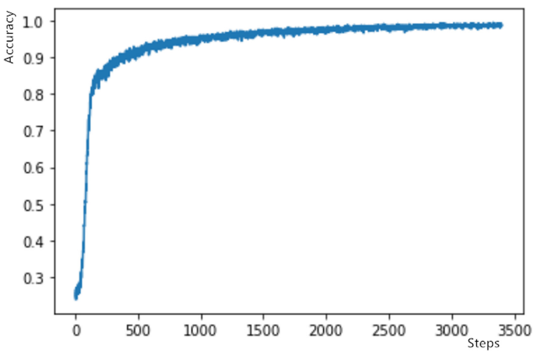


Fig. 10. Accuracy of training steps (MAML, SVHN, 5-way 5-shot)

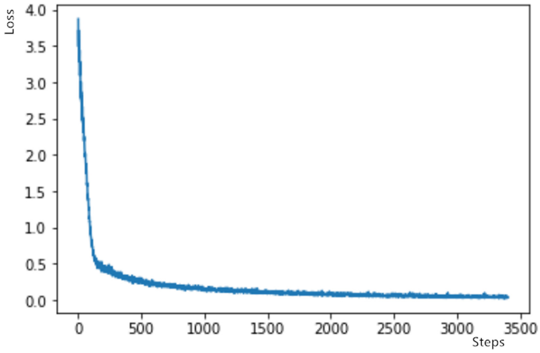


Fig. 11. Loss of training steps (MAML, SVHN, 5-way 5-shot)

Table 3. Test accuracy (MAML, SVHN)

Test accuracy						
Steps	500	1000	1500	2000	2500	3000
Test accuracy (%)	2-way 1-shot learning					
	51.7	58.1	65.9	68.8	70.3	69.6
	2-way 5-shot learning					
	77.1	80.5	80.2	79.7	79.9	80.3
	5-way 5-shot learning					
	56.6	54.6	52.9	52.4	50.5	50.6

In order to show clearly the performance of MAML for few shot learning with SVHN dataset, we refer to the results of MAML for few shot learning with Omniglot and MiniImagenet dataset [1]. In addition, we also apply the MAML model to the MNIST dataset as a reference (Table 4).

Table 4. Test accuracy with different dataset (MAML, 5-way 5-shot)

Dataset	Omniglot	MiniImagenet	SVHN (ours)	MNIST (ours)
Test accuracy	$99.9 \pm 0.1\%$	$63.11 \pm 0.92\%$	$53.6 \pm 3\%$	$72.4 \pm 1.4\%$

Among these results, the experimental accuracy of Omniglot dataset is much higher than that of other datasets. The accuracy of MNIST dataset and MiniImage dataset is the second and third, while the accuracy result of SVHN dataset is the lowest. This proves that the relatively simple and less interference images have a higher experimental accuracy.

5 Comparison with Siamese Network

In order to compare the performance of different few shot learning methods for SVHN dataset, we also use Siamese network to carry out 5-way 5-shot learning for SVHN dataset.

5.1 Siamese Neural Networks

Siamese network [4] is still based on convolutional neural network, but unlike traditional CNN, the input data is paired (x_1 and x_2). In these pairs of data, if the two kinds of data are the same, label the pair of data as 1, if the two kinds of data are different, label the pair of data as 0.

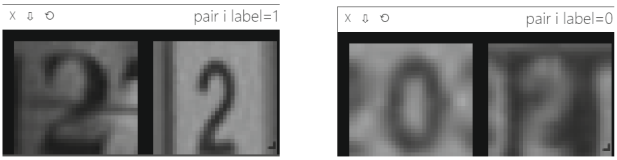


Fig. 12. Examples of pairs (Siamese). The two pictures on the left have the same category, so the label is 1; the two pictures on the right have different categories, so the label is 0.

The two images in a pair of data will be input into the same CNN respectively, and the output data will be merged by Euclidean distance, and the resulting data will be classified into 0 (different categories) or 1 (same category). Correspondingly, the loss function should be binary cross entropy and the optimizer is Adam.

5.2 Experiment

For the experiment of 5-way 5-shot, we extract 5 kinds of images from the SVHN dataset, and extract 5 images from each kind as the support set. Then, with the first category of the support set as the reference, we extract 5 * 5 pictures from the SVHN dataset as the test set. In this way, the first five of these data pairs should have a prediction value of 1 (the same picture type), and the next 20 should have a prediction value of 0 (different picture types).

For the test after each iteration, if the predicted results show that the first five pairs of data have the highest similarity, then the test is regarded as a prediction success. The experimental results are shown in the figures below.

From the figures, the accuracy of Siamese network for 5-way 5-shot learning with SVHN dataset can reach 47.63%. But the range of its accuracy is relatively large. On average, the accuracy of Siamese network is between 40% ± 5%.

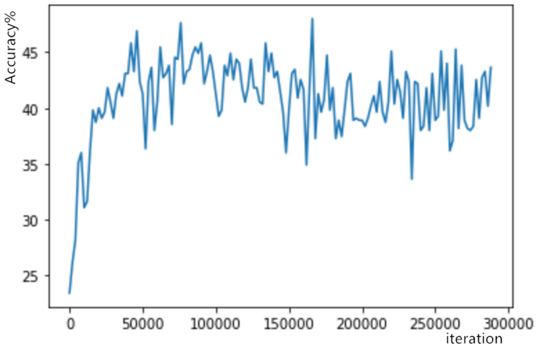


Fig. 13. Test accuracy of Siamese net (SVHN, 5-way 5-shot)

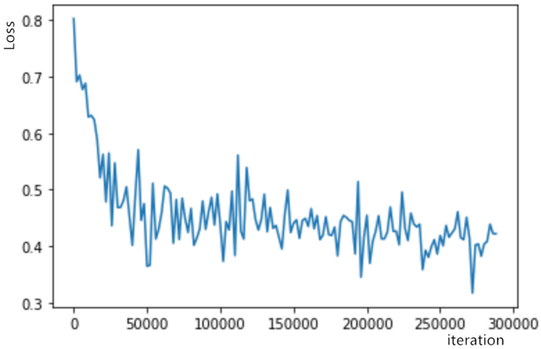


Fig. 14. Loss of Siamese net (SVHN, 5-way 5-shot)

Table 5. Test accuracy of Siamese (SVHN, 5-way 5-shot)

Steps	60000	120000	180000	240000	300000
Test accuracy (%)	45.4	41.8	37.2	38.3	39.2

5.3 Comparison

In our experiment, it took only 6150 s for Siamese network to carry out 300000 iterations, and achieved relatively stable data, while it took 9522 s for MAML method to complete 1500 iterations to achieve stable data., The accuracy of 5-way 5-shot experiment with MAML method is 5 to 10% points higher than that of Siamese network.

In addition, it can be seen from the figures that the test accuracy and loss range of Siamese network is relatively large and from the perspective of percentage error (Table 7), the percentage error of MAML is 5.6% and the percentage error of Siamese network is 12.5%. This means that the generalization performance of MAML network is better, and MAML has better stability when dealing with various types of few shot learning tasks.

In order to further verify the difference between MAML and Siamese network, we apply Siamese network to MNIST dataset, and the experimental results are shown in Table 6, Table 7 and Fig. 15. In MNIST dataset, the advantages of MAML model are more obvious. The accuracy of test is 73%, while the Siamese network is only 59%. Moreover, the percentage error of MAML is lower than that of Siamese network.

In Table 7, we also record the performance of traditional CNN model in 5-way 5-shot learning. For the simple MNIST dataset, CNN has a $48.6\% \pm 6.1\%$ accuracy rate, but for the complex SVHN dataset, CNN's performance is much worse than the other two methods dedicated to few shot learning. In order to show the performance of the three methods more intuitively, we also summarize the test accuracy of the three methods and draw Figs. 16 and 17.

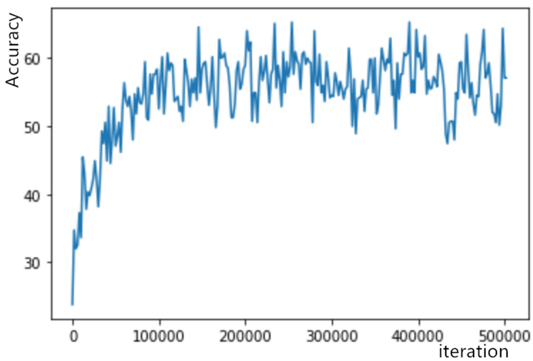


Fig. 15. Test accuracy of Siamese net (MNIST, 5-way 5-shot)

Table 6. Test accuracy of Siamese (MNIST, 5-way 5-shot)

Steps	60000	120000	180000	240000	300000
Test accuracy (%)	54.8	54.3	60.1	64.1	59.8

MAML and Siamese are two different ways to solve the problem of feed shot learning. The goal of MAML is to get the best initial parameters of generalization performance, while Siamese is trying to judge the similarity of two images to

Table 7. Comparison of MAML, SIAMESE and CNN (5-way 5-shot)

Comparison of MAML, SIAMESE and CNN (5-way 5-shot)			
SVHN			
	MAML	SIAMESE	CNN
Steps	1500	300000	500
Time spent(s)	9522	6150	579
Test accuracy	$53.6\% \pm 3\%$	$41.8\% \pm 5\%$	$20.8\% \pm 1\%$
Percentage error of test accuracy	5.6%	12.5%	4.8%
MNIST			
Steps	1500	300000	500
Times spent(s)	8015	4929	554
Test accuracy	$72.4\% \pm 1.4\%$	$57.2\% \pm 7\%$	$48.6\% \pm 6.1\%$
Percentage error of test accuracy	1.9%	12.2%	12.5%

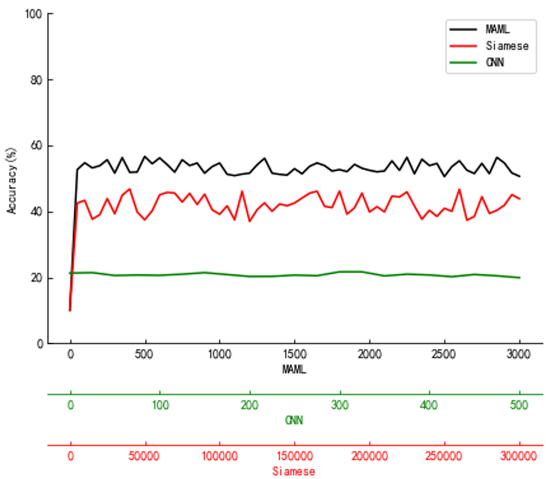


Fig. 16. Test accuracy of 5-way 5-shot of different methods (SVHN). The black line represents MAML, the red line represents Siamese network, and the green line represents traditional CNN. (Color figure online)

classify. They are all good research directions, but in experiments based on SVHN and MNIST datasets, MAML has better performance, both in terms of test accuracy and generalization performance. Meanwhile, from the experimental results of these two methods for SVHN dataset, the accuracy of the test still has a large space to improve.

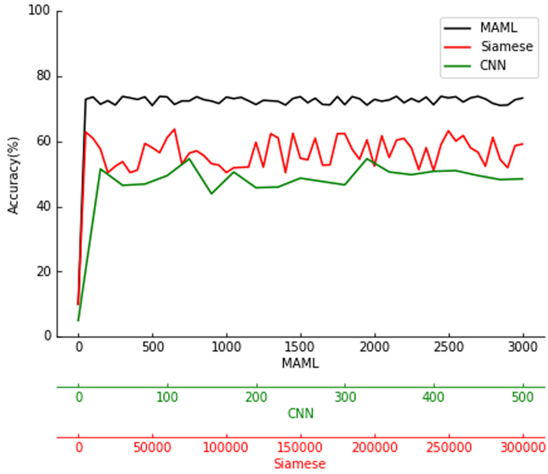


Fig. 17. Test accuracy of 5-way 5-shot of different methods (MNIST). The black line represents MAML, the red line represents Siamese network, and the green line represents traditional CNN. (Color figure online)

6 Conclusions and Future Works

In this paper, we introduce the algorithm of MAML and apply it to SVHN dataset based few shot learning. Our experiments include 2-way 1-shot, 2-way 5-shot and 5-way 5-shot, and the accuracy of these three experiments are about 70%, 80% and 50%, respectively. The accuracy will decrease with the increase of image category and the decrease of sample number. For 5-way 5-shot learning, we compared the accuracy of MAML application on different datasets: the accuracy based on simple datasets is significantly higher than that based on complex datasets. Also, we also verify that for SVHN and MNIST dataset, MAML has better performance than Siamese network.

In addition, from the experimental results, we find that the training accuracy of each experiment will eventually approach 100%, and the test accuracy will be stable at a specific value, which verifies that the datasets with fewer categories can also be applied to the few shot learning experiments, but in the process of training, the same kind of image data has a certain positive effect on the model parameters, and the training accuracy may be biased, and the type of test accuracy is different from the training part, so the test accuracy is credible.

For the future work, in order to prevent the above problem, SVHN dataset can be synthesized into images of multi digit numbers, and each number includes several images. Moreover, the influence of the value of the MAML hyper parameter on the experimental performance is worth further discussion.

References

1. Finn, C., Abbeel, P., Levine, S.: Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, pp. 1126–1135 (2017). JMLR.org
2. Goodfellow, I.J., Bulatov, Y., Ibarz, J., Arnoud, S., Shet, V.: Multi-digit number recognition from street view imagery using deep convolutional neural networks. arXiv preprint arXiv:1312.6082 (2013)
3. Hochreiter, S., Younger, A.S., Conwell, P.R.: Learning to learn using gradient descent. In: Dorffner, G., Bischof, H., Hornik, K. (eds.) ICANN 2001. LNCS, vol. 2130, pp. 87–94. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44668-0_13
4. Koch, G., Zemel, R., Salakhutdinov, R.: Siamese neural networks for one-shot image recognition. In: ICML Deep Learning Workshop, Lille, vol. 2 (2015)
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
6. Nichol, A., Achiam, J., Schulman, J.: On first-order meta-learning algorithms. arXiv preprint arXiv:1803.02999 (2018)
7. Normalization, B.: Accelerating deep network training by reducing internal covariate shift. corr.-2015. Vol. abs/1502.03167. http://arxiv.org/abs/1502.03167 (2015)
8. Ravi, S., Larochelle, H.: Optimization as a model for few-shot learning (2016)
9. Santoro, A., Bartunov, S., Botvinick, M., Wierstra, D., Lillicrap, T.: One-shot learning with memory-augmented neural networks. arXiv preprint arXiv:1605.06065 (2016)
10. Snell, J., Swersky, K., Zemel, R.: Prototypical networks for few-shot learning. In: Advances in Neural Information Processing Systems, pp. 4077–4087 (2017)
11. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. In: Advances in Neural Information Processing Systems, pp. 3630–3638 (2016)
12. Xia, T.: Research on classification method of small sample image based on meta learning. Master's thesis, University of Electronic Science and technology (2019)
13. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_53