Exploring the Performance of Recurrent Neural Networks on the Recognition of Manipulated Images

ZiJian Li

Research School of Computer Science, Australian National University
u6152831@anu.edu.au

Abstract. With the situation of unknown to the factors which can influence the cognitive ability to manipulated images, this paper is for exploring whether recurrent neural networks are helpful for studying the pattern of whether humans can perceive manipulated images through different length of observing duration with different levels of attention, we train neural networks with a dataset which included the 372 records from Sabrina Caldwell's experiment [4]. We find that all tested networks are able to solve the problem. However, comparing to recurrent neural network and long short term memory network, bidirectional recurrent neural network is more worthwhile to recommend.

Keywords: manipulated images, cognitive ability, recurrent neural networks

1 Introduction

1.1 Background

The quantity of image information increases rapidly with each passing day. According to its different states and characteristics, image information conveys diversified information to the public [3]. So, the image manipulation is always a topic in the process of gathering information due to its quantity and importance. Because the manipulation directly controls the content of image which is also the source of information. With the control of the information spreading, manipulating technique could even cause the social, economic, and political threats [6]. So, it is useful to figure out whether people could realize the manipulation, for mastering the deviations in given image information.

1.2 Purpose

This research will mainly focus on exploring whether RNN type neural networks are helpful in analyzing the relationship between human eye reading behavior and the ability to recognize manipulated pictures. The RNN type neural networks are RNN, Long Short Term Memory networks (LSTM) and Bidirectional Recurrent Neural Network (BiRNN). The direction of exploration includes parameter adjustment for RNN, the characteristic analysis of RNN, and similar RNN based neural networks.

1.3 Dataset

The data set for this study was collected from the Sabrina Caldwell's experiment [4]. The experiment recorded the data of eye gaze when participants recognized whether the image was manipulated or not. The image observation data of the subjects in the data set includes the fixation times and duration of the whole image and the manipulated area. There is also the number of participants and the number of images they observe in the dataset, as well as the binomial value of whether the observed images are manipulated. The output of the dataset is the individual judgment of the participants on whether the observed image is manipulated. The data used in this study is only part of Sabrina Caldwell's experiment, with only 372 sets of data.

1.4 Relative Research

With the purpose on verifying the suitability of training recurrent neural network (RNN), here was already a research [7] proved two preliminary conclusions on the relationship between human eye reading behavior and the ability to recognize manipulated pictures. It says that the well-trained full connected three-layer feedforward neural network (FC3NN) can reach 70% accuracy on predicting the relation. However, the research also shows that BiRNN has a worse performance comparing with the FC3NN, which means it is necessary to implement the comprehensive investigation on BiRNN for figuring out the reason.

Another relative research showed that the native photograph pictures with the manipulations have a higher rate to lead failure to detect substantial changes [1]. With the pictures in Sabrina Caldwell's experiment [4] are about nature, the pattern should be more difficult to be trained. So, the accuracy of the trained neural network may not very high, and it is possible to treat the 70% accuracy as the standard. For the RNN, if its accuracy reaches 70% with the reasonable parameters and other performance, we can judge it is at least acceptable for this dataset and the purpose.

Normally speaking, increasing the attention of the manipulated area in the manipulated image (fixation, duration of eye gaze, which could be find in gathered dataset) can improve the recognition ability of the manipulated image [4]. So, the trained neural network is supposed to be meaningful. The accuracy of trained models should be greater than 50%, which is the natural probability of guessing true or false.

1.5 Further Benefits

This research facilitates the following:

- Exploring the suitability of RNN to the relationship between human eye reading behavior and the ability to recognize manipulated pictures.
- Providing the basis and used parameters for other similar research of training neural network on the dataset of eye gaze.

2 Method

The method is divided into three main parts and serval processing part. The first main part is about the construction of BiRNN and LSTM, including the required parameter settings, technique usages and reasons. The second part is the evaluation criteria of neural network, which is used to modify the parameters. The third part is to simply explain the adjustment mode and basis of the parameters in the experiment, and the stages of the results to the experiment will need to be analyzed.

2.1 Neural Network Construction with Determined techniques and parameters

To explore the comprehensiveness, we constructed three neural networks, namely simple RNN, LSTM, BiRNN. These three neural networks are RNN type or RNN derived type. Therefore, the parameters and technical types that we set for these neural networks are basically the same among the three neural networks. Therefore, the design and construction of parameters will not be written separately in the following description. The initial value of each parameter may or may not be given, because there may be great changes in the following discussion, and only the initial idea will be given in this section.

Data preprocessing.

Starting with data preprocessing, the data cleaning and data reorganization are same as the preliminary research [7]. We fix the format of the column title and create the 'correctness' column as the target column to represent the correctness of identifying manipulated pictures. Then, we choose the input features and target column (see Table.2), and we could also compare it with original data (see Table.1).

statistics	participant	num_fixs	fixs dur	num_man_fixs	man_fixs_dur	image	image manipulated	vote
mean	39.69086	81.846774	18.944414	16.064516	3.777149	11.97043	0.491935	0.432796
Std	22.453633	38.000932	10.322925	19.913996	5.211553	1.40913	0.500608	0.57654
min	1	6	0.967	0	0	10	0	0
1-quarter	20	54	11.86375	1	0.14975	11	0	0
median	40	76	17.1755	8	1.532	12	0	0
3-quarter	58.25	108	24.75425	23	5.5985	13	1	1
max	80	215	63.459	115	32.305	14	1	2

Table. 1. The statistical data of original dataset, which include mean value, standard deviation, minimum value, value of the first quarter of data, median value, value of the first 3 / 4 data and maximum value.

statistics	num_fixs	fixs dur	num_man_fixs	man_fixs_dur	image_manipulated	correctness
mean	81.846774	18.944414	16.064516	3.777149	0.491935	0.596774
Std	38.000932	10.322925	19.913996	5.211553	0.500608	0.491206
min	6	0.967	0	0	0	0
1-quarter	54	11.86375	1	0.14975	0	0
median	76	17.1755	8	1.532	0	1
3-quarter	108	24.75425	23	5.5985	1	1
max	215	63.459	115	32.305	1	1

Table. 2. The statistical data of processed dataset for training and testing, which include the same type of data as Table.1. The last column is the target and the other columns are inputs.

5-Fold Cross Validation.
There are two main reasons for choosing the k-fold cross validation method. One is that the k-fold cross validation method is more stable than other validation methods in small data sets. The other is that by implementing the cross validation method, the number of tests for each parameter configuration scheme can be appropriately reduced, which means that the total duration of the experiment can be reduced and the efficiency can be improved.

To the reason of setting the k value as 5, it mainly caused by the size of the dataset. In my opinion, there are only 372 available data in the dataset, and the test set cannot be too small under the premise of ensuring sufficient training set size. So, I chose at least 20% as the test set to reduce the bias between the test set data obtained in k = {1, 2, 3, 4, 5} training.

Batch Size.
The size of the batch is also determined by the size of the dataset. If the batch size is too small, the model from each step of training may produce some large biases, and some of the bias may not be corrected in subsequent training. Combining with the situation that the dataset used in this study is very small, the batch size of both the test set and the training set is directly set to the size of the corresponding dataset

Fixed Structure of Neural Network.
According to the data set for training and testing (see Table.2), the number of input features is 5. Because the target output column is 'correctness', and the data in this column contains only two values. So, the number of output classes is 2. As for the number of hidden layers, due to the small size of the data set, the number of input features is only 5. In addition, the number of hidden layers is directly set to 1. Such a setting can also improve the training efficiency to a certain extent.

There are many schemes to determine the number of neurons in the hidden layer, and most of them follow the empirical rule. So, it is better to consider the number of neurons in the hidden layer as a variable. Its domain will be discussed later in the next section.

Activation function.
There is not too much selection of activation function in RNN type neural network. Compared with the 'relu' series activation function, I chose 'tanh' activation function because the 'relu' series activation function requires high learning rate control. With 'tanh' activation function, the only thing we need to pay attention to is the offset caused by the fact that the activation function is not symmetrical about the center of the zero point. In this study, we do not need to focus on the problem of gradient disappearance which may cause by 'tanh' activation function in other cases, because the number of epochs and hidden layers are very small. Before the gradient disappears, the neural network will be trained or overfitting. Due to the purpose of this study, it is not necessary to find the extreme optimal solution. So, the 'tanh' activation function is appropriate.

Loss function.
The choice of the loss function is 'crossEntropyLoss', because for binary problems, cross entropy loss function is available.

2.2 Neural Network Construction with Result-Related variables

Number of hidden neurons.
When the number of input features, output classes and hidden layers are fixed, we can discuss the number of hidden neurons. Because there are many empirical rules to determine the number of hidden neurons, the number is not easy to determine. It is better to set the number of hidden neurons as variables and determine the range to construct different neural networks. We can set the lower limit of interval as the number of input features. Since the number of input features does not contain each other, at least corresponding or more hidden layer neurons are needed to train. For the upper limit of interval, we do not need to determine it. What we should do is increasing the number by 1 every time and comparing the performance of neural networks. If the performance converged the suitable hidden neuron number can be found.

Learning rate, epoch, weight decay.

The reason why these three variables are put together is that they all play a role in judging overfitting and local minimum value. Starting with local minimum value, without the simulated annealing algorithm, if we want to jump out of the local minimum, we can adjust the training parameters of the neural network. We can also use the random gradient descent method, but this method will be left in the optimizer section to explain. For the adjustment of training parameters, these three parameters have the higher degree of freedom than others. So, we need to try different combinations of these three parameters. To the overfitting, the experimental strategy is to set the weight decay to 0 first, that means the L2 regularization is not used temporarily. After the combination of learning rate and epoch reach the best (a low enough loss) and the neural network will be overfitting. Now we could increase weight decay to avoid overfitting and get the result.

Optimizer.

In terms of optimizer, Adam is the first choice in terms of universality and performance. Considering that the model may fall into the local optimal solution, the neural network may need an SGD optimizer to help the model get rid of the local optimal solution. So, the optimizer selection range is {Adam, SGD}.

2.3 Evaluating Methods

The evaluation part of neural network mainly consists of two parts: network performance evaluation and rationality evaluation. To evaluate the performance of the neural networks built for classification, accuracy is a key factor. Because the accuracy directly represents the ability to solve problems

The rationality evaluation partially relies on accuracy. The ideal situation is that the accuracy of training set and test set is very close when they converge, and the trend of loss is the same. If there is a big difference between the accuracy of training set and test set, it can be divided into the following two situations:

- When the accuracy of the test set is far greater than that of the training set, it is impossible to explain with common sense, so it is necessary to check the code and check the cause of the error.
- When the training set accuracy is much higher than the test set, first check the overfitting problem, then check the code.

The loss changes of test set and training set can be summarized into four categories:

- If the loss of test set and training set increases at the same time, the neural network fails, and the code needs to be checked.
- If the loss of test set and training set decreases at the same time, the initial judgment of training is normal, which needs to be judged according to the specific situation.
- If the loss of the test set increases and the loss of the training set decreases, the neural network may over fit, and further observation and parameter adjustment are needed.
- If the loss of the test set decreases, the loss of the training set increases, and the neural network fails. First, we need to check whether the training set and the test set are confused.

2.4 Simple Experimental Process

The core idea of the experiment is to find a reasonable and accurate RNN type neural network model. So, parameter adjustment is also based on the evaluation method. The order of finding the optimal value of variables is determined. For all three neural networks, the priority is to find the number of hidden neurons. The optimizer needs to be decided later. Based on the decision of the optimizer, the accuracy of the model can be determined by the learning rate, the exploration of epoch and weight deck. If there is an accident in the middle of the experiment, there will be a simple discussion about whether to analyze the accident. If necessary, a new method may be inserted for further research.

The phased results will be presented according to the necessity of the conclusion description, and the process data will not appear in the next results and discussion section.

3 Result and Discussion

The Result and Discussion section will focus on the staged results of the three neural networks and the discussion based on the results.

3.1 RNN

Hidden neuron number.

To find the number of hidden neurons suitable for RNN, we need to set the initial value of other variables for comparison. Other variables are set as follows. Learning rate equals to 0.001, epoch equals to 1000, weight decay equals to 0.001, optimizer is Adam. So, we could get the result in Table.3

hidden nueron num	5	6	7	8	9	10	11
training accuracy (%)	70	69	70	71	73	72	73
testing accuracy (%)	63	63	64	65	65	64	63

Table. 3. The average accuracy records with different number of hidden neurons for RNN. All the accuracy records are the approximate mean value over 20 times running. The value settings of other variables are the same for every time training.

When the number of hidden neurons is 9, the accuracy of training set and test set has reached the peak. So, the number of hidden neurons is determined to 9.

Optimizer.

When selecting the optimizer, the loss function image needs to be referenced. However, due to the small data set, the loss function image of the test set is usually irregular, so it is often unable to judge the universality of the training results. Moreover, in many cases, because the test set is not extensive, the neural network will have a overfitting phenomenon at the beginning of training. Therefore, under the premise of k-fold cross validation, when the loss of test set is increasing or keeping stable from beginning in any fold, the fold cannot be recorded as an effective training. The final training results will be calculated from the rest of the normal training results. For example, Fig.1:

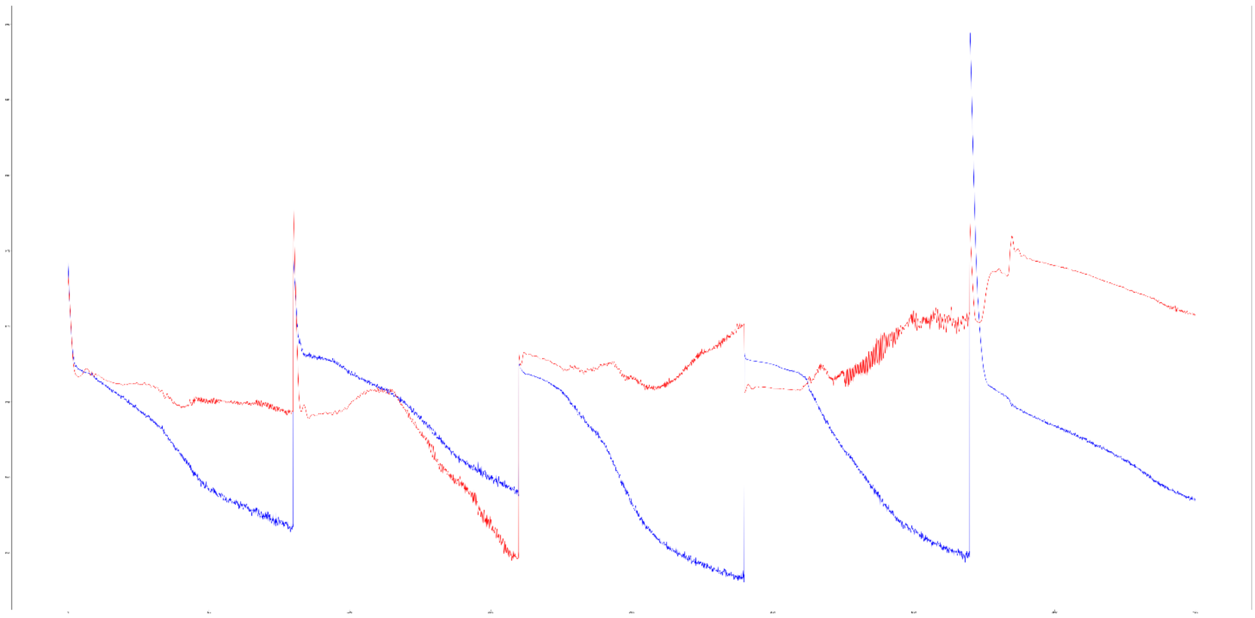

Fig. 1. This is a change process of RNN loss by training with 5-fold cross validation. The blue scatter is the loss of training set and the red scatter is the loss of test set. The Y axis records the loss values which are in the range of 0.5 (bottom) to 0.90 (top), and X axis records number of epochs. The epoch number of each training is 800. So, the domain of X is [1, 4000]. The learning rate is 0.001, the weight decay is 0.01, and the selected optimizer is Adam.

After RNN training with Adam optimizer, the accuracies of training set and test set are 72.039% and 65.201%. In the third and fourth training, there is obviously overfitting phenomenon, so the results of these two times should be discarded. The accuracy of RNN training set and test set was 70.973% and 68.905% after the two results were omitted, which is also an acceptable performance. Obviously, it is more reasonable to ignore the overfitting to calculate the accuracy, and this also helps to evaluate the bias between the training set and the general test set.
If the SGD optimizer is selected for training, although there will not be much overfitting phenomenon, other problems in trained RNN with SGD optimizer will occur (see Figure 2).

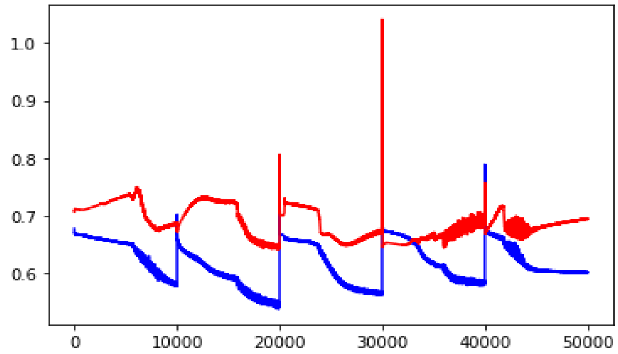

Fig. 2. This is a change process of RNN loss by training with 5-fold cross validation. The blue scatter is the loss of training set and the red scatter is the loss of test set. The Y axis records the loss values which are in the range of 0.55 (bottom) to 1.00 (theoretical top), and X axis records number of epochs. The epoch number of each training is 10000. So, the domain of X is [1, 50000]. The learning rate is 0.001, the weight decay is 0.001, and the selected optimizer is GSD.

Although the loss curve in Figure 2 is not so bad, it can be observed that the number of epochs is large enough to make RNN jump out of the local minimum. However, no matter how RNN is trained, the loss difference between training set and testing set will only increase with the deepening of training. In Figure 2, the accuracy of the training set is 70.656%, while the accuracy of the test set is 63.635%. Compared with the result of full-connected 3-layer neural network in the preliminary experiment, GSD optimizer performs worse in the multiple RNN training of current experiment. So, GSD seems not suitable for RNN, and the optimizer of RNN is chose as Adam.

Learning rate, weight decay, epoch.

Because the total amount of data set is too small, it is difficult to find a always perfect parameter combination in RNN for each repeated experiment. As the final purpose of the experiment is to test the suitability of RNN network for the relationship between human eye reading behavior and the ability to recognize manipulated pictures, the measurement method can be changed as follows. When the number of epochs is enough to make RNN overfit, the minimum value of learning rate in the general initial range is selected, so the learning rate is 0.001. Set the weight decay to 0 for testing. If the neural network does not show acceptable performance before RNN overfitting, increase the value of weight decay until acceptable performance appears. If the value of weight decay is too large to make RNN quickly overfit and there is not any good performance, then RNN is not suitable for the current experiment.

The experimental results are as follows:

- When weight decay = 0, the reasonable performance appears in 600-1200 epoch, otherwise it will never occur.
- When weight decay = 0.0001, there are three types of performance. The case 1 is directly overfitting, the case 2 is the best performance in the whole process, and the case 3 is the accuracy gap between testing set and training set is always maintained at 8% - 10%. The probability of the three situations is almost the same
- When weight decay = 0.001, the probabilities of case 1 and case 2 were increasing. The probability of case 2 is 47%
- When weight decay = 0.01, the probabilities case 2 were decreasing. So, the testing should be stopped

In all acceptable RNN performances, the accuracy of training set is between 70% and 72%, with the accuracy of testing set is within plus or minus 5% of accuracy to training set. All the features above are gathered by observing the accuracy changing in training process 20 times. Because of the 5-fold cross validation, the observed neural networks are 100. Then, here is a part of the original data in Table.4 to show the process of observing.

```
Epoch [201/2000], Step [1/1], Loss: 0.6314  Accuracy: 64.9660 Test Accuracy: 57.6923
Epoch [301/2000], Step [1/1], Loss: 0.6193  Accuracy: 64.9660 Test Accuracy: 57.6923
Epoch [401/2000], Step [1/1], Loss: 0.6074  Accuracy: 66.3265 Test Accuracy: 62.8205
Epoch [501/2000], Step [1/1], Loss: 0.5931  Accuracy: 67.3469 Test Accuracy: 65.3846
Epoch [601/2000], Step [1/1], Loss: 0.5782  Accuracy: 71.0884 Test Accuracy: 69.2308
Epoch [701/2000], Step [1/1], Loss: 0.5638  Accuracy: 72.4490 Test Accuracy: 67.9487
Epoch [801/2000], Step [1/1], Loss: 0.5559  Accuracy: 73.4694 Test Accuracy: 66.6667
```

Table. 4. This is a part of original data used in finding the acceptable performance. The loss value and the accuracy are belonging to the training set with the epoch in the front. The test accuracy is belonging to the testing set tested by the under-training neural.

After training, RNN can also meet the standards in the preliminary study [7]. But for the problem that can be solved by general neural network, it is unnecessary to use RNN. So, the conclusion is that although RNN is a kind of neural network that can be used, there will be simpler neural network construction for the same problem, so RNN is not suitable for the current problem.

3.2 LSTM

The study of LSTM will follow the process of RNN study. If there is no special case, the whole process will not repeat large paragraphs from previous section of RNN.

Hidden neuron number.

With the same setting, learning rate is 0.001, epoch is 1000 and weight decay is 0.001. The results in Table.5 are also approximate mean values over 20 times experiments.

hidden nueron num	5	6	7	8	9	10	11	12	13
training accuracy (%)	71	72	73	73	75	76	77	77	78
testing accuracy (%)	64	64	64	64	64	64	65	63	62

Table. 5. The average accuracy records with different number of hidden neurons for LSTM. All the accuracy records are the approximate mean value over 20 times running. The value settings of other variables are the same for every time training.

There is also the peak we can follow, which the hidden neurons number is 11. However, the gaps between training accuracy and testing accuracy is larger than the gaps from RNN.

Optimizer.

The simulation with Adam is basically the same as what the result is in RNN, except that in the interval between training set and testing set. The gaps in LSTM is generally larger than the gaps in RNN.

To the GSD optimizer, in the process of judging whether the GSD optimizer is suitable (see Table.6), we found that the performance of the model in accuracy has reached the required level, while the parameters still have room for adjusting, which the current parameters setting are in the formula (1).

$$\text{parameter setting} = \{lr = 0.001, wd = 0.00001, ep = 30000\} \tag{1}$$

Where the lr is learning rate, wd is weight decay, ep is epoch.

```
Epoch [16001/30000], Step [1/1], Loss: 0.5707  Accuracy: 71.5254 Test Accuracy: 66.2338
Epoch [17001/30000], Step [1/1], Loss: 0.5619  Accuracy: 72.5424 Test Accuracy: 67.5325
Epoch [18001/30000], Step [1/1], Loss: 0.5557  Accuracy: 73.2203 Test Accuracy: 68.8312
Epoch [19001/30000], Step [1/1], Loss: 0.5521  Accuracy: 73.2203 Test Accuracy: 70.1299
Epoch [20001/30000], Step [1/1], Loss: 0.5498  Accuracy: 73.2203 Test Accuracy: 71.4286
Epoch [21001/30000], Step [1/1], Loss: 0.5481  Accuracy: 73.5593 Test Accuracy: 70.1299
Epoch [22001/30000], Step [1/1], Loss: 0.5467  Accuracy: 73.5593 Test Accuracy: 70.1299
Epoch [23001/30000], Step [1/1], Loss: 0.5454  Accuracy: 73.8983 Test Accuracy: 70.1299
```

Table. 6. This is a part of original data used in judging the optimizer. The loss value and the accuracy are belonging to the training set with the epoch in the front. The test accuracy is belonging to the testing set tested by the under-training neural.

So, the LSTM is suitable for using in studying the purpose. However, it takes a lot of time to train LSTM with current parameters, further parameter optimization may be needed, but it is sure to be effective.

3.3 BiRNN

In the cited paper [7] it is mentioned that in the current problem of neural network model training, BiRNN does not perform well because the accuracy distance between the testing set and the training set is too large. However, there is not any data to prove it. So, with the concept of ‘bidirectional’ [5] this part mainly uses data to observe the performance of BiRNN.

Hidden neuron number.

The method to judge the number of hidden neurons is not different with the methods we used in previous two sections. There is also a peak in Table.7 at the number of hidden neurons is 10. When it comes to 11, although the training accuracy is increasing, the testing accuracy starts to decrease. So, the number of hidden neurons should be 10.

hidden nueron num	5	6	7	8	9	10	11
training accuracy (%)	75	75	78	79	79	79	80
testing accuracy (%)	65	64	64	63	63	64	62

Table. 7 The average accuracy records with different number of hidden neurons for LSTM. All the accuracy records are the approximate mean value over 20 times running. The value settings of other variables are the same for every time training.

Optimizer.

In the same way as LSTM and with the same parameter settings, we can directly find the condition that the training accuracy of BiRNN exceeds 70% in Table.8.

```
Epoch [9001/30000], Step [1/1], Loss: 0.5713  Accuracy: 68.3849 Test Accuracy: 67.9012
Epoch [10001/30000], Step [1/1], Loss: 0.5639  Accuracy: 70.7904 Test Accuracy: 71.6049
Epoch [11001/30000], Step [1/1], Loss: 0.5542  Accuracy: 70.1031 Test Accuracy: 74.0741
Epoch [12001/30000], Step [1/1], Loss: 0.5441  Accuracy: 71.4777 Test Accuracy: 72.8395
Epoch [13001/30000], Step [1/1], Loss: 0.5382  Accuracy: 72.5086 Test Accuracy: 72.8395
Epoch [14001/30000], Step [1/1], Loss: 0.5333  Accuracy: 72.5086 Test Accuracy: 71.6049
```

Table. 8. This is a part of original data used in judging the optimizer. The loss value and the accuracy are belonging to the training set with the epoch in the front. The test accuracy is belonging to the testing set tested by the under-training neural.

And the training results (see Table.9) also show that the training is successful, because the testing accuracy is also in an acceptable range. Although the parameters still need to be adjusted, the results show that BiRNN can be used in this study. BiRNN training is faster than STLM, so it can meet the conditions earlier in around 10000 epoch in Table.8.

training set			testing set		
confusion matrix	predicted_true	predicted_false	confusion matrix	predicted_true	predicted_false
actual_true	68	49	actual_true	25	8
actual_false	23	151	actual_false	16	32
training_accuracy = 75.26%			testing_accuracy = 70.37%		

Table. 9. This is a table of confusion matrixes for training set and testing set. The accuracies of both sets are given.

4 Conclusion

In the experiment, we constructed RNN, LSTM and BiRNN to determine whether RNN neural network is helpful for analyzing the relationship between human eye reading behavior and the ability to recognize manipulated pictures. For the dataset from Sabrina Caldwell's experiment [4], RNN is able to use, but due to the limitation of data set, it is not very clear about the universality. Therefore, for the purpose of this paper, RNN type neural network is worth using. But for small data sets, we need to ensure the stability of the results when we use them. In general, BiRNN has better performance than the other two neural networks because of its convergence efficiency and stability.

In future study, we still need to explore the neural network model with higher accuracy. At the same time, it is important for us to collect data from the investigation, not other's experiment. Trying to figure out more input features and is also way to build a high accuracy model. Here is a more advanced research paper [2] realize the recognition of the manipulated image, which can be used as a reference for future research in this field.

Reference

1. Ball, P., Elzemann, A., BUSCH, N.A.: Interueniunt et invisibilium: experimentorum Manipulating imagines in in SCENA mutatio ignorantiam recti et memoria. Res Behav XLVI, 689-701 (MMXIV). https://doi-org.virtual.anu.edu.au/10.3758/s13428-013-0414-2
2. Bayram, S., Avcibas, I., Bülent Sankur, Ovid, N. D.: Image manipulation deprehendatur. Journal of Electronic Imaging. XV (IV). 041102. (MMVI)
3. Caldwell, S., Gedeonis, T, R. Jones, Copernicus, Nicolaus, l.: Minus perfecte sapie: Theoria Motus autem fundati oculis conspicias investigatione De Humanum perceptiones et Manipulated Unanipulated Digital Images. Edita Consilio Congresso de mundo, et in Rei Electricae inuestigandae Systems and Computer Scientia (MMXV)
4. Ionescu, N.: Online politica communication: ad imaginem eget partes in facebook. Revista De romana & Communication Jurnalism Si comunicare Romanian Journal ex Media (I) (MMXIII)
5. Nejad, A. F., Gedeonis, C. D.: Et retiacula Bidirectional neural prototypes genus. ICNN'95 - vol.3 1322-1327 Networks.pp Neural Congressus Internationalis (MCMXCV).
6. A global metus referre 12th ed. Mundus Oeconomicarum Forum (MMXVII)
7. Li. Z .: Neural Lorem Network: et inveniens Lectio Book Status Sagaciter inter Exemplum distinctionis Manipulated Images (MMXX)