Discriminating Real from Fake Smiles Using Pupillary Responses and Recurrent Neural Network

Minzhe Chen[1 [u6660082]]

Research School of Computer Science, Australian National University u6660082@anu.edu.au

Abstract. Pupillary response can show whether the smile is real or fake, and neuron network can extract unseen features from the data. In this work, a recurrent neural network (RNN) is proposed to discriminate real from fake smiles. The input is pupillary size data of both left and right eyes, labels of real or fake smiles. Then a RNN is constructed, and it can predict whether smile is real or fake according to pupillary size. Multiple pre-processing methods, network structures, loss functions and optimization methods are tried to improve its performance. Network reduction technique is also used to remove the useless neurons. The result is evaluated by its confusion matrix and average accuracy of cross-validation. The final accuracy of the classification model can reach 70%. Considering the difficulty of discriminating real from fake smiles and limited data instances, this performance is good.

Keywords: Recurrent Neural Network · Classification · Network Reduction · Cross-validation · Real Smile · Fake Smile · Pupil Size.

1 Introduction

1.1 Smile and Pupil Size

Human smiles can express various emotions [3]. Real and fake smiles look similar but may show different feelings. Therefore, distinguishing fake from real smiles is a significant task, and many researchers attempted to tell the difference. Valstar et. al. [9] uses smiling videos to discriminate real from fake smiles. Then Ochs, Niewiadmoski and Pelachaud [8] discovered the smiling characteristics. Recently research used Smiles data set and discovered a strong relationship between pupil size and smile, and states that pupil size will change in different way depending on whether the smile is real or fake [7]. In my last project, I chosed the first version of Smiles data set, where there are labels and average pupil sizes. The data is pre-processed and the length of vectors are fixed. I used pupil size data to train a classifier, then such a classifier can predict whether the smile is true or fake given a sequence of the pupil sizes. And the accuracy can reach 70%. In this project, I uses the new version of Smiles data set, where there are pupil sizes of each eye, and the length of each data sample is unknown. The label of data is added manually according to the description in the paper [7].

1.2 Recurrent Neural Network and Sequence Learning

Neural network usually has great performance on classification tasks. Folkes, Lahav and Maddo(1996) [4] successfully trained a classifier on galaxy spectra. Later, more relevant techniques and more complicated structures are developed to improve the performance. RNN has feedback connections, so its inputs can be a sequence of unknown length. Moreover, RNN can memorize previous context, and use previous information to generate output. Therefore, it can be used in this project, because the length of each data instance in Smile data set is unknown, and RNN can discover the change of pupil sizes. On the other hand, RNN is not stable because gradients can explode or vanish when backpropagating gradients through long time windows. To solve this issue, Hochreiter and Schmidhuber [6] introduced Long Short-Term Memory (LSTM) to RNN. In my project, LSTM is also used as a part of my model.

Encoding and network reduction will make the model more accurate and efficient. Therefore, I propose a neuron network to distinguish real from fake smiles using pupil sizes from Smile data set as input. I have attempted several methods of data pre-process, input/output encoding, and different architectures of my model. Then improve the accuracy and effectiveness by encoding and network reduction techniques [5]. Finally, I evaluate, analyse and report the model by cross-validation and confusion matrix. And experiment demonstrates that RNN and LSTM can work on this project, but its efficiency and accuracy is not so good as expected.

In conclusion, the main contributions of this project are:

1. Train a RNN model that take pupil sizes as input and predict whether the smile is real or fake.
2. Use LSTM structure to make the model can be trained through backpropagation.
3. Use network reduction technique to remove unnecessary hidden units to make the model efficient.

2 Method

2.1 Model Structure

The input of model is a sequence of pupil sizes of both eyes, the length of the input is unknown. Then it go through RNN and a linear layer, and finally the model will produce a real number between 0 and 1 as output to predict a real smile(1) or fake smile(0). This model contains five parts:

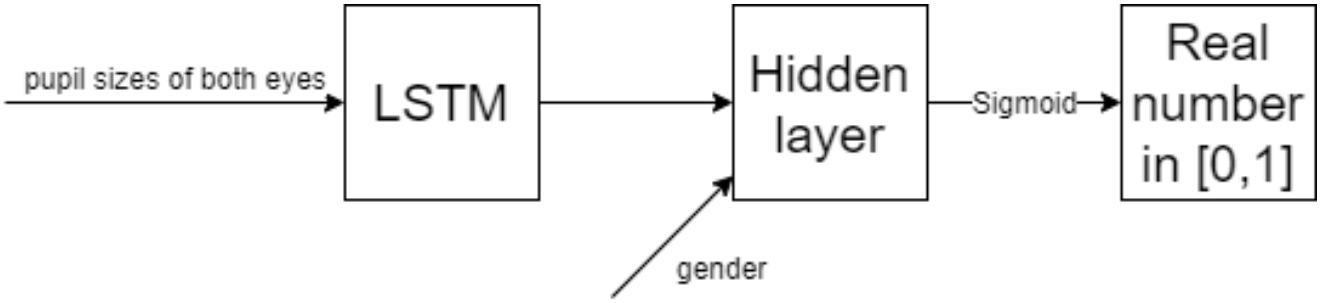

Fig. 1. Model structure, there are 5 neurons in the hidden linear layer, neurons are fully connected.

1. Load and pre-process data
2. Input encoding
3. RNN
4. Hidden Layer processing
5. Output and predict a class

The neuron network contains a LSTM and a output layer. Sigmoid function is used as activation function in the output layers. The structure of the network is shown in the figure 1.

During training this model, network reduction technique are used to remove unnecessary hidden units through computing and compare the similarity and functionality of them. In addition, cross-validation and confusion matrix are used to evaluate the performance.

2.2 Data Pre-processing and Input Encoding

Several steps were used in data pre-processing and input encoding. Firstly, as there are many NaN values in the original data, I changed them 0 to make each entry of the data set meaningful. Moreover, I added a feature called gender manually according to the excel table, as male and female has difference pupil response [7].

For input encoding, Bustos and Gedeon [1] suggested that the structure of the original data should not be destroyed. Therefore, I used 0 and 1 to encode the labels which is 'Fake' and' Real', and used (1,0) and (0,1) to encode the gender feature.

2.3 Recurrent Neural Network and Long Short Term Memory

Because of the unknown length and relationship of pupil sizes, I chose RNN as my model. My RNN contains feedback connections and can memorize and learn from previous input to generate output. However, RNN was found to be not stable when backpropagation was used. Therefore, I used a LSTM [6] to overcome this issue. It takes the pupil size vectors and previous h and c as input, and generate an output and h, and memorize c.

There are one hidden layer and one output layer in this model. The hidden layer contains 3 hidden neurons and the output layer is one output neuron. All the neurons are fully connected. Hidden layer uses sigmoid function as activation function. This is because sigmoid function can produce a number between 0 and 1, and its derivative is large when the input is closed to 0.5, which is great for classification between two classes [2]. In conclusion, the network works according to equations below:

$$output = Sigmoid(LSTM(leftsize, rightsize)) \tag{1}$$

where leftsize and right size represent the input data. Output represent the output vectors from the model. LSTM means the RNN model described above and Sigmoid() is sigmoid funtion:

$$Sigmoid(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

It is also significant to determine which activation function should be used. I tested some combinations of different activation functions and different number of layers and units. For this two-class classification model, sigmoid function can produce a real number between 0 and 1, which represent the soft predicted class. Moreover, the performance when ReLU function is used is much worse. I believe that this is because negative values are also meaningful in this model. Therefore, I use sigmoid activation functions.

2.4 Training Details

Loss Function This model uses L1Loss as loss function. Because the output and ground truth values are scalars, the L1Loss is equal to the absolute value of the difference between predicted and ground truth values:

$$Loss = ||y_{pred} - y||_1 = |y_{pred} - y| \tag{3}$$

where y_{pred} and y represent the predicted and ground truth values respectively.

I have tried some other loss functions including mean square loss (MSE) and binary cross entropy loss (BCELoss), their performance is no better than L1Loss.

Optimizer I use ADAM as the optimizer. Stochastic gradient decent (SGD) is more efficient than other optimizer like Adam, but Adam has a better performance when the network is deeper.

Learning Rate and Training Epochs I set my learning rate equal to 0.01 and number of epochs equal to 200, because my computer could not work with too many epochs.

2.5 Network Reduction Technique

Removing unnecessary hidden units can improve the efficiency of the neural network. Gedeon and Harris [5] proposed a network reduction technique to discover and remove the useless hidden unit. This technique detect and compute the distinctiveness and similarity of hidden units, and remove the units which has a similar of opposite function with another hidden unit. The units with unreasonable output will also be removed. Accordingly, the distinctiveness of my hidden units is computed, and some of them whose angle is less than 15 degree or larger than 165 degree are removed. Afterward, the efficiency of the network significantly increased, but the accuracy of the network became less stable.

2.6 Cross-Validation and Confusion Matrix

It is necessary to use cross-validation to make full use of the data. Therefore, I split the data into 10 folders randomly, use one folder as test set and the other as training set, train the model each time and test the average accuracy. In this way, the outcome becomes less sensitive to data choice, which reveals the real accuracy of the model.

The accuracy is computed with confusion matrix. The matrix shows that false positive and false negative are balanced, which means the model makes reasonable decisions.

3 Results and Discussion

This result section will discuss not only the final result but also the initial result and how the performance of the network improved. Cross-validation is used to generate the result. The final result is evaluated by computing the confusion matrix and mean accuracy. To calculate the accuracy, I ran the model 3 times and report the average performance.

3.1 Initial Accuracy and Improvement

Initially, the row input and simplest RNN were used and the accuracy was approximately 50%. Then I add gender as a feature because the pupil size of man and woman change differently [7]. This new feature significantly improved the testing accuracy to about 60%. Next, I changed the number of hidden layers and used Adam as the optimizer. Those improvement slightly increases the accuracy. After that, loops are used to find the best structure of hidden layers. Since my laptop can hardly work when the number of hidden layers and the number of layers is high, I mainly reported the result of using only 2 layers after LSTM.

I also tried some more complex network structures which contains three hidden layers or more. However, since the LSTM is more computational expensive and I have only my laptop with limited computation source, they work extremely slow and show similar performance.

As shown in the table 1, the best result is produced by an one layer neural network with 5 or 3 hidden units. To achieve better results, I used 5 neurons in the layer. As a result, the accuracy of my model becomes 66% on average. After adjusting some other hyper-parameters like the number of epoch, the accuracy of cross-validation is closed to 70%.

Table 1. The accuracy of 2-layer network after LSTM with different number of hidden units, when the number of neurons of second layer is 0, there is only one hidden layer, the best result are bold

1st hidden layer	2nd hidden layer			
number of units	0	1	2	3
1	0.57	nan	nan	nan
2	0.63	0.56	nan	nan
3	**0.66**	0.59	0.56	nan
4	0.60	0.60	0.52	0.52
5	**0.69**	0.60	0.44	0.52

3.2 Network Reduction Technique

To improve the efficiency and performance of the neural network, I implement network reduction technique [5] to detect useless hidden neurons by computing its distinctiveness. I start with 10 hidden neurons, and found that several pairs has an angle below 15 degree of larger than 165 degree, which means they are similar or opposite. Therefore, I remove 5 of them and recompute the functionality of the remaining 5 hidden neurons as shown in the table3. The angles of each pairs of the hidden units are listed in the table2. In this way, all 5 hidden neurons output meaningful outputs, each has a high output variance, which means that they are not similar to other hidden units or to itself.

With network reduction technique, the efficiency of the neuron network is improved and the unnecessary hidden units are removed. However, it hardly improves the accuracy of the model and it makes the prediction less stable. It is believed that this technique is not fit to a network with only one hidden layer. If there were more layers, we could remove some units depending on how many useless units are there in the layer.

Table 2. Angles between each pair of 20 hidden units

	0	1	2	3	4
0	0.0	80.0	150.0	75.0	109.0
1	80.0	0.0	90.0	18.0	44.0
2	150.0	90.0	0.0	89.0	51.0
3	75.0	18.0	89.0	0.0	39.0
4	109.0	44.0	51.0	39.0	0.0

Table 3. The output of 5 hidden units, each column represents a unit and each row is a testing data simple

	0	1	2	3	4
0	0.36	0.65	0.83	0.90	0.94
1	0.22	0.69	0.94	0.90	0.98
2	0.88	0.68	0.20	0.83	0.38
3	0.83	0.69	0.19	0.86	0.44
4	0.54	0.68	0.65	0.95	0.88

3.3 Final Result

The final result is computed using cross-validation as shown in table 4. I train the model and predict several times and sum all together. The average accuracy is approximately 75%. Considering the limitation of data samples, this performance is satisfying. Fake smiles are hard to predict, because it has unclear pupil size changes [7]. Therefore, the accuracy of predicting fake class is about 66% while the accuracy of predicting real class is more than 90%. The accuracy of male and female is also reported, because they have different pupil responses and female should be hard to predict[7]. The results show that the accuracy has significant difference between male and female. The reason should be the unclear pattern of female's smiles. Another possible reason is that the data samples of female is less than that of male.

Table 4. Confusion Matrix. The result is the sum of 3 times cross validation

		Actual	
		F	T
predicted	F	268	26
	T	146	292

4 Conclusion and Future Work

This project can produce accurate classification of fake or real smiles. The relevant researches of this topic provide important patterns that can be used for classification. This is because there are interesting relationships between smile and pupil sizes. RNN is an effective method to discover the changes of pupil sizes along time, and LSTM, which contains memory and forget gates, can help RNN model work. Avoiding over-fitting is significant in this project because the training data is limited. Cross-validation and early stop can be helpful, because it can make full use of the training data. Network reduction technique [5] enhances the efficiency of the model by removing unnecessary hidden units. The accuracy might be improved more computational resource is available, because LSTM is computational expensive so the number of training epochs need to be reduced. In this way, the loss cannot converge to the minimum.

The future work can focus on extract more features from the original videos using convolutional neural network (CNN). Those new features may improve the accuracy because using pupil size as the only feature is not robust and easy to over-fit. Moreover, more hidden layers can be used if there are more training data. In this way, the complex model can discover more useful patterns and results in better performance. Network Reduction technique might be uses in multiple layers and LSTM. Another possible future work is to use both new and old version of smile data set, because version 1 contains average values and version 2 is more detailed.

References

1. Bustos, R., Gedeon, T.: Decrypting neural network data: a gis case study. In: Artificial Neural Nets and Genetic Algorithms. pp. 231–234. Springer (1995)
2. Deisenroth, M.P., Faisal, A.A., Ong, C.S.: Mathematics for machine learning. Cambridge University Press (2020)
3. Ekman, P., Friesen, W.V.: Felt, false, and miserable smiles. Journal of nonverbal behavior **6**(4), 238–252 (1982)
4. Folkes, S., Lahav, O., Maddox, S.: An artificial neural network approach to the classification of galaxy spectra. Monthly Notices of the Royal Astronomical Society **283**(2), 651–665 (1996)
5. Gedeon, T., Harris, D.: Network reduction techniques. In: Proceedings International Conference on Neural Networks Methodologies and Applications. vol. 1, pp. 119–126 (1991)
6. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8), 1735–1780 (1997)
7. Hossain, M., Gedeon, T., Sankaranarayana, R., Apthorp, D., Dawel, A.: Pupillary responses of asian observers in discriminating real from fake smiles: A preliminary study. In: Measuring Behavior. pp. 170–176 (2016)
8. Ochs, M., Niewiadomski, R., Pelachaud, C.: How a virtual agent should smile? In: International Conference on Intelligent Virtual Agents. pp. 427–440. Springer (2010)
9. Valstar, M.F., Gunes, H., Pantic, M.: How to distinguish posed from spontaneous smiles using geometric features. In: Proceedings of the 9th international conference on Multimodal interfaces. pp. 38–45 (2007)