Neural Networks to the Question: Are You Really Angry?

Weiqi Chen[1],

School of Computer Science and Engineering
Australian National University
u5423569@anu.edu.au

Abstract. Facial emotion analysis nowadays has played an essential role in machine learning field. Researcher have conducted several experiments to classify the acted and genuine anger. In this paper, a 2 layer of shallow neural network and a long short-term memory network were proposed to perform binary classification between the acted and genuine anger. In terms of the 2-layer neural network, magnitude, brute force and functional measures were applied to determine the importance of the input neurons. The network was then adjusted based on the outcome of the three measures. With regards to the LSTM model, various hyperparameters combinations were used to conducting trials on the model. The overall performance of LSTM model was better than the neural network model's, which achieved accuracy. Techniques and analysis such as sensitivity analysis with aggregated ranking and genetic algorithm could be applied to determine the optimized hyperparameters.

Keywords: facial expression, neural network, binary classification, input significance, LSTM

1 Introduction

For several decades, facial emotion recognition has widely raised people attention as it is a medium to understand the intentions of others during communication. Meanwhile, with the development of machine learning techniques, numerous researches related to facial expression have been conducted to explore the potential applications to human-computer interaction. The most famous facial expression categorisations proposed by Ekman [3] recognised 6 basic emotions including anger, disgust, fear, happiness, sadness and surprise. One of the early algorithms that contributes to facial recognition is Nearest Feature Line, which extract two feature points on a person's face and determine if a pass a particular feature line [7]. However, this method only gave an insignificant performance; whereas machine learning techniques such as shallow neural network, convolutional neural network and recurrent neural network have been established and lead to a better performance especially for the classification task [5].

According to M. Mather, M.R. Knight [2], it is easier for human beings to detect angry faces in comparison with upset and happy faces; therefore, facial expression of anger analysis could be used as a mechanism in respond to threats for cue. Several experiments have been conducted to determine the differences between acted and genuine anger in past decades. One of the experiments carried out by Chen [1] has proved that that pupillary response patterns could have better prediction of veracity with respect to anger compared with verbal respond in general. The experiment invited 22 participants to view two types of anger stimuli which were last for around 1 to 2 seconds. The videos were then regarded as the benchmark for the classification. Meanwhile, the corresponding verbal and pupillary responds of each participant was record by answering questions afterwards and eye tracker at frequency of 60 Hz. The results illustrated that the accuracy of classifying the acted and genuine anger by human verbal respond can only achieve 60%, whereas by applying ensembles of machine learning classifier, the performance could have an accuracy above 90%.

In this paper, a simple two layers neural network is firstly trained to classify the acted and genuine anger. The dataset used for this neural network is the statistical summary from Chen's experiment with 6 input features. The neural network is then adjusted based on the outcomes of magnitude analysis, brute force and functional measure on input neuron importance as proposed in [6]. Meanwhile, a deep learning technique, long short-term memory (LSTM) recurrent neural network is established to perform the same binary classification task but with the raw time-series dataset of Chen's study. Discussion with regards to these two models is provided.

2 Method

2.1 Two Layers Neural Network

The task for the shallow neural network is to perform a binary classification between acted and genuine anger. The initial network was built in the structure of 6-10-2, being 6 input, 10 hidden and 2 output neurons. The 6 inputs are the mean, standard deviation, two differences between the left and right pupil responds, and the first and second principle components of the pupillary response over the time course; while the output neurons represent the acted and genuine anger classes. The total size of the dataset 400 and it was split into training/test set with ration of 0.8.

The network was designed to be feedforward, which means all the connections between neurons do not form a cycle and only flow in the forward direction from the input nodes, hidden modes to the output nodes. Each node is connected to the next layer nodes with a unique weight. Since it is a classification task, the activation function chosen for the output is the basic sigmoid logistic function given below:

$$y = \frac{1}{1 + \exp(-x)} \tag{1}$$

The neural network was trained by error backpropagation using Adam optimizer with a 0.01 learning rate. As there are only two classes, the error measure was selected to be the cross-entropy loss calculated as:

$$Loss = \frac{1}{N} \sum_{I}^{N} y_i \log\big(p(y_i)\big) + (1 - y_i) \log\big(1 - p(y_i)\big) \tag{2}$$

The network training is terminated once it hits 500 epochs. The performance of the neural network is measured by the test dataset.

2.1.1 Magnitude measures

For the purpose of magnitude measure, the is used to rank the 6 inputs. Wong, Gedeon and Taggart (1995) [8] applied the following measure to calculate the average contribution of a node in a layer to a node in the next layer:

$$P_{ij} = \frac{|W_{ij}|}{\sum_k^n |W_{ik}|} \tag{3}$$

The contribution of an input neuron to an output neuron therefore could be written as:

$$Q_{ij} = \sum_{r=1}^{nh} (P_{ij} \times P_{jk}) \tag{4}$$

For comparison purpose, the contribution of an input neuron to the 2 output neurons were averaged to rank the importance inputs to the binary classification the network is solving.

2.1.2 Brute Force Analysis

Brute force could be referred as a trial and error-based search algorithm that instead of finding the shortcut to improve the performance, it searches all the possible combinations and compared the performance to find the best solution. In this paper, the brute force was applied to try out all the combination of inputs. The significance of inputs was determined based on the average test loss values among all the combinations. According to [7], eliminated 1 input result in inconsistent results, hence 2 inputs were eliminated at the same time. As there were 6 input features, a total number of 15 combinations should be tested. Four networks with the same topology were trained for each combination with same initial weights, which were generated for the full input network, and the weight matrix was adjusted by excluding the corresponding inputs.

2.1.3 Functional measure

Proposed by Gedeon and Harrsion (1991) [2], the technique of distinctiveness could be applied for the functional measure as indicated below:

$$angle(i, j) = \tan^{-1}\left(\sqrt{\frac{\sum_{p}^{pats} sact(p,i)^2 * \sum_{p}^{pats} sact(p,j)^2}{\sum_{p}^{pats} (sact(p,i) * sact(p,j))^2} - 1}\right) \tag{5}$$

The angel between two input, i and j was used to determine the similarity. The formula was further adapted by alternating the input representation by the pattern of input to hidden weight:

$$sact(p, h) = norm(weight(h)) - 0.5 \tag{6}$$

The ranking of the inputs was calculated based on the average similarity of each input to the rest.

2.2 LSTM Recurrent Neural Network

Recurrent neural network is a machine learning technique that could work highly efficient with sequential inputs or outputs of both such as time series data. However, it also suffers from the short-term memory as it could only contain limited information from earlier times steps. Long short-term memory, referred as LSTM is a special type of RNN, which replace the nonlinear units in RNN by the memory blocks. Each memory block contains several internal memory gates that can regulate the flow of information. A basic unit of LSTM is shown in []. The raw dataset from Chen's study is time-series pupillary response unlike the statistical summary used for shallow neural network, it could be suitable to apply LSTM neural network as it is capable of learning long-term dependencies.

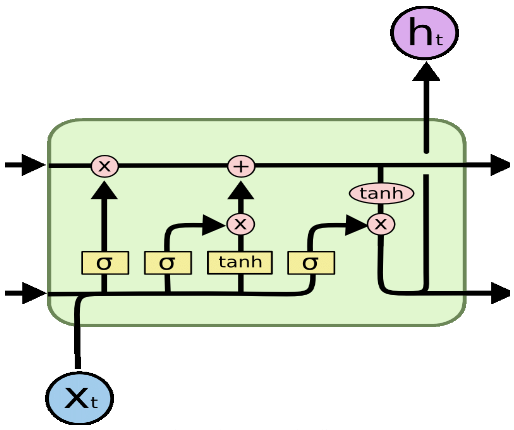

Fig. 1 Basic Unit of LSTM

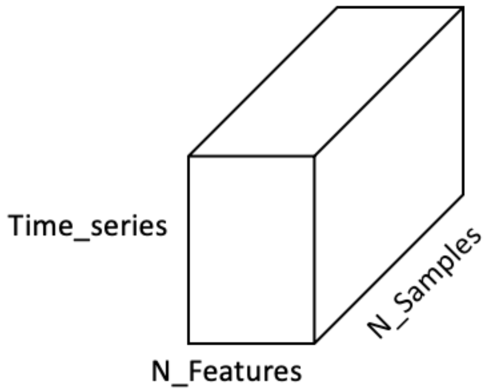

Fig. 2 Data Structure

2.2.1 Data Preprocessing

The raw dataset consists of 22 participants' left and right pupillary response with regards to 10 acted anger videos and 10 genuine anger videos. However, the dataset also missed some participants data; therefore, the first step of pre-processing data is to remove the empty columns. As a result, the total sample size has been reduced to 390 with 2 input features. Observations were captured by eye gaze tracker at 60 Hz (i.e. 60 data points per second). However, since the length of each video is different, the input sequence length is also various. For the shorter videos, dummy padding with zeros is applied to ensure the lengths of the input sequences are in line with the maximum number of the recorded frame. The whole dataset was then reshaped to 3D structure as shown in figure 1, which is the required data structure in LSTM Keras deep learning library. To be more specific, one sample is a window of the time series data, each window has the same number of times steps, and a time step has two features while samples are stacked alone with the third axis.

2.2.2 Two Layer LSTM

The proposed model has a single LSTM hidden layer with 100 units follower by a dropout layer with ratio 0.2 whose intention is to prevent overfitting by randomly ignoring some of the neuron while retaining model accuracy. Before the final output layer with SoftMax as activation function to make classification, a fully connected layer is used to interpret the extracted features by the LSTM hidden layer.

Similar to the 2-layer neural network, it was trained by error backpropagation using Adam optimizer with the errors calculated by cross-entropy function. Batch size and the number of epochs are treated as two adjustable hyperparameters when training LSTM. Batch size refers to the number of samples that feed to the model each time so that the model can distinguish the common patterns and features across the samples. While epoch is the number of the passing the whole dataset through the model. With the increase of epochs, the model might result in underfitting to overfitting. Therefore, with the optimal values of these two hyperparameters, the performances of the model could improve significantly.

2.3 k-Fold Cross Validation

Both of the models are adjusted based different conditions. In order to further to test the reliability of the adjusted neural network, k-Fold cross validation was carried out.

K-fold cross validation is simply a resampling procedure. The dataset was divided into k subset of data; while k-1 subsets were used for training and 1 subset was used for testing. The whole procedure could iterate K times implying there should be k testing results. The advantage of such methods is the randomly generated subset data could be both repeated used for training and testing purposes, which means the results could be verified once during each iteration and it might prevent overfitting [4].
The performances were compared with the original neural network and provided in next section.

3 Results and Discussion

3.1 Outcomes of the 2-Layer Neural Network

In the following table, the significance of inputs calculated from the three different measures in terms of magnitude, brute force and functional are provided.

Table 1. Results of three measures

	Most Significant					Least Significant
Magnitude (M)	2	6	1	3	5	4
Brute Force (B)	3	6	1	2	5	4
Functional (F)	3	6	1	2	4	5

As can be seen from the table above, the input ranking of the three methods are similar. Both of the magnitude and functional measures are 60% in accord with the brute force ranking, which agree that the second and third most significant inputs are the second principle components and the mean respectively. However, with regards to the most and least important inputs, it did not indicate any convergency but one of them was still in line with the brute force.

The graph below indicates the cross-entropy loss of the four neural networks with regards to the 15 different combinations of 4 inputs neurons. It should be noticed that the order did not sort by the index of the corresponding combination but the magnitude of the loss. As can be seen from the figure, there was discontinuity in the error which indicates there should be some sudden degradation of the neural network. This could also imply the existence of some specific inputs might heavily impact the performances of the neural network.

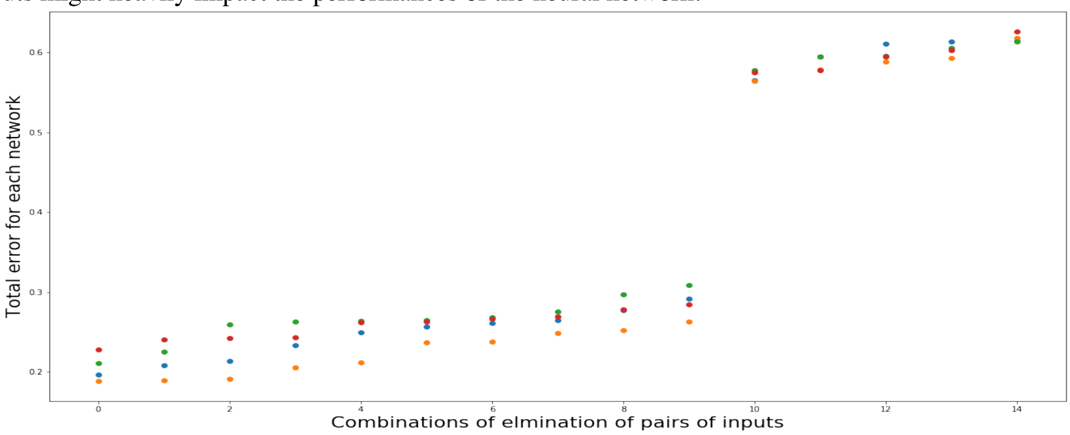

Fig. 1 Errors of different combination

Since both the magnitude measure and brute force analysis agreed on that the 4^{th} input is the least significant neuron, it has been removed from the neural network. Meanwhile as the 2^{nd} and 3^{rd} inputs have the highest ranking, the initial weight on these two neurons have been increased by 0.01. The performance of the adjusted neural network indeed displayed an increase on the test accuracy and the total correct prediction number. Additional measurements are also provided. In terms of the recall, it was a static value represents the ability to correctly classify the acted anger; while precision indicates the proportion of the test dataset has identified the acted anger were actually acted anger. Although there was a decrease on recall, precision has raised with regards to removing the least important input, which also fits the hypothesis that there was a trade-off between recall and precision. Therefore, f1 score measure has also introduced here to indicate whether the model has found an optimal blend. As can be seen, the f1 score also grew implying removing the 4^{th} input could improve the neural network performance. The results and related measure are shown in the Table 4.

Table 2. Comparison between the base and adjusted neural network

	Train Loss	Train Accuracy	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
Base	0.2098	91.54%	79.71%	55	90.00%	71.05%	79.41%
Remove 4^{th} input and increase the weight	0.1614	89.73%	82.61%	57	90.00%	75.00%	81.82%

Furthermore, the neural network has been further adjusted by eliminating the two least important inputs, the 4^{th} and 5^{th} neuron, which were identified by the three measures.

Table 3. Results of Eliminating pairs of inputs.

	Train Loss	Train Accuracy	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
Base	0.2098	91.54%	79.71%	55	90.00%	71.05%	79.41%
Remove (4^{th}, 5^{th}) input	0.5900	68.28%	43.48%	30	40.00%	36.36%	38.10%

As displayed by the table, eliminating even the pairs of the least significant input neurons could also severe destroy the network performance especially in terms of the test accuracy and correct prediction number. The overall outcome of the adjusted network was given above, whose accuracy has been increased to nearly 82%. The following figure illustrate the relationship between loss and the total number correct. The combination of anti-correlation and total correct prediction number could be a suitable indication of the degree of difficulty of the classification problem for the network.

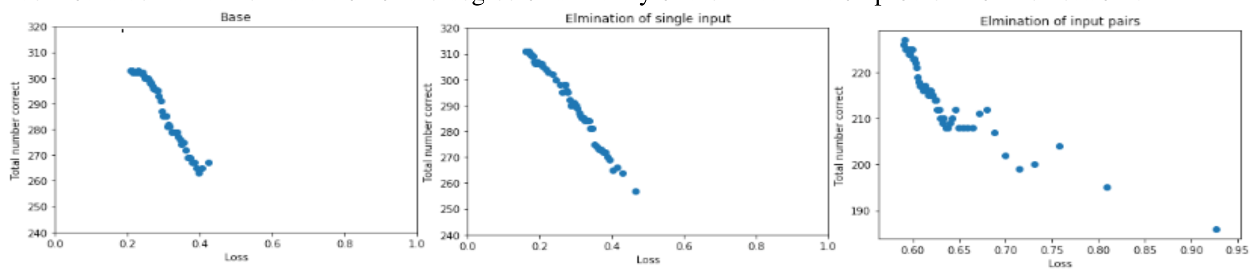

Fig 3. Correlation between loss and total number correct of three different networks

In order to further to verify the reliability of the adjust network, 5-fold cross validation was conducted as to maintain the comparability to the 0.8 train/test dataset. The average performance was summarized as below:

Table 4. 5-fold Cross-Validation outcomes

	Train Loss	Train Accuracy	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
Average of Cross-Validation	0.3605	84.75%	81.00%	65	78.24%	83.32%	80.14%
Base	0.2098	91.54%	79.71%	55	90.00%	71.05%	79.41%

It was noticed that the average cross-validated result indicates that the adjusted neural network has a better performance overall, and there was a significant difference in the recall and precision trade-off. In short, the cross validation has shown that the adjusted neural network has improved the performance compared with the original neural network, but the accuracy could still be further increased compared to Chen's results.

3.2 LSTM Model

In this section, the performances of LSTM model with various hyperparameters were provided. The initial model was trained with 128 batch size and 250 epochs which were commonly used for initially established the network.

Table 5. Outcomes of Initial Model

	Batch Size	Number of Epochs	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
Base	128	150	86.76%	59	96.67%	78.38%	86.56%

The model was firstly experiment with different number of epochs range from 50 to 350 with fix batch size of 128.

Table 6. Outcomes of alternating number of epochs

Number of Epochs	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
50	83.12%	64	90.00%	75.00%	88.31%
150	88.31%	68	87.50%	88.61%	93.51%
250	93.51%	72	88.63%	93.98%	91.91%
350	90.91%	70	84.62%	97.06%	90.41%

With the increase on the number of epochs, it can be seen that the model has been over trained with number of 350 since the overall performance has been drop. As a result, the numbers of epochs should not excess 350. In order to further validate, a graph that includes both train loss and validation loss was provided below, where there was a significant validation loss between the 300t to 350.

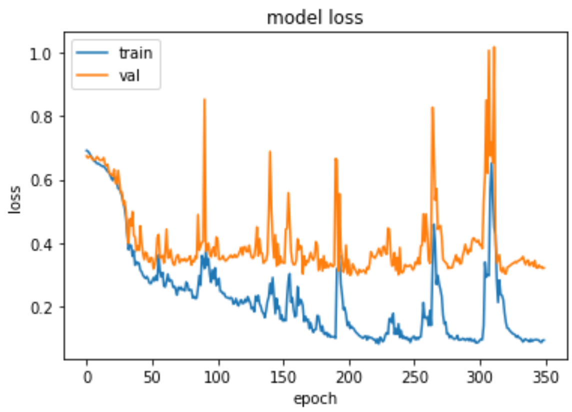

Fig 4. Model Loss

The model was then trained with different batch size values that were dividable by 8. The results were presented in table 7.

Table 7. Outcomes of alternating number of batch size

Batch Size	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
64	85.90%	67	91.48%	85.33%	89.74%
128	89.74%	70	87.50%	92.11%	84.62%
256	87.17%	68	87.50%	87.50%	87.50%

Taking all of the results presented above into consideration, the model performance indeed could be enhanced by increasing the two hyperparameters' values to a reasonable range. However, it should be noticed that these two values should not be tested and determined independently as the model was affected by the combination of these two values. There are numerious potential combinations that might exhibit better performance. A test on number of epochs and batch size with 250 and 128 was conducted. In other to verify the reliability of such model, cross validation was carried out. The results were shown below. As can be seen, the performance has been boosted and the accuracy was close to the outcome illustrate in Chen's paper.

Table 8. Outcomes of alternating number of batch size

	Batch Size	Number of Epochs	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
Base	128	150	86.76%	59	96.67%	78.38%	86.56%
Average of Cross Validation	128	256	91.02%	71	91.80%	90.91%	90.89%

In short, both of the models have been successfully built and were able to classify the fake and genuine anger. For the 2-layer neural network, it has been adjusted by elimination an input neuron which leads to an increase on accuracy at around 80%. With regards to the LSTM model, by adopting different hyperparameter, the model could exhibit a better performance compared to the neural network in general.

Table 9. Outcomes of Initial Model

Average of Cross-Validation	Train Loss	Train Accuracy	Test Accuracy	Correct Prediction Number	Recall	Precision	F1-Score
2-Layer NN	0.3605	84.75%	81.00%	65	78.24%	83.32%	80.14%
LSTM Model	0.3019	79.81%	91.02%	71	91.80%	90.91%	90.89%

4 Conclusion and Future Work

In conclusion, a shallow neural network has been adjusted to provide a satisfying overall performance on a binary facial expression classification task. Magnitude measure, brute force analysis and functional measures were conducted to investigate the significance of the input neuron. Based on the three measures, the original 2-layer neural network were adapted by eliminating different combination of input neurons and alternating the weight on specific input neuron. The neural network that had the most satisfying overperformance has removed the least significant input neuron while increase the weight on the most important neurons.

Further techniques and analysis such as sensitivity analysis with aggregated ranking technique could still be implemented to verify to the ranking provided by the three measures mentioned above. Meanwhile, genetic algorithm could be implemented to determine the optimization hyperparameter of LSTM in terms of the number of hidden neurons, learning rate, number of training epoch and batch size etc.

References

1. Chen, L., Gedeon, T., Hossain, M. Z., & Caldwell, S. (2017, November). Are you really angry? detecting emotion veracity as a proposed tool for interaction. In Proceedings of the 29th Australian Conference on Computer-Human Interaction (pp. 412-416). ACM

2. Gedeon, TD and Harris, D "Network Reduction Techniques," Proceedings International Conference on Neural Networks Methodologies and Applications, AMSE, vol. 1, pp. 119-126, San Diego, 1991.

3. M. Mather, M.R. Knight, Angry faces get noticed quickly: threat detection is not impaired among older adults, J. Gerontol. Ser. B: Psychol. Sci. Soc. Sci., 61 (2006), 54-57. DOI: https://doi.org/10.1093/geronb/61.1.P54

4 N. Li, F. He, W. Ma, R. Wang and X. Zhang, "Wind Power Prediction of Kernel Extreme Learning Machine Based on Differential Evolution Algorithm and Cross Validation Algorithm," in IEEE Access, vol. 8, pp. 68874-68882, 2020.

5. P. Ekman, "Strong evidence for universals in facial expressions: a reply to russell's mistaken critique.," 1994.

6. P. R. Dachapally, "Facial emotion detection using convolutional neural networks and representational autoencoder units,". arXiv: 1706.01509, 2017.

7. T.D. Gedeon, Data mining of inputs: analysing magnitude and functional measures, Int. J. Neural Sys. 8 (2) (1997) 209–218.

8 Verma, A., Singh, P., & Alex, J. S. R. (2019). Modified Convolutional Neural Network Architecture Analysis for Facial Emotion Recognition. 2019 International Conference on Systems, Signals and Image Processing (IWSSIP). doi: 10. 1109/iwssip. 2019. 8787215

9. Wong, PM, Gedeon, TD and Taggart, IJ "An Improved Technique in Porosity Prediction: A Neural Network Approach," IEEE Transactions on Geoscience and Remote Sensing, vol. 33, n. 4, pp. 971-980, 1995.