Genuine Anger Detection through Standard Neural Network and Recurrent Neural Network

Erni Gao

Research School of Computer Science, Australian National University, Canberra, Australia

U6740613@anu.edu.au

Abstract. Human beings are innately good at computerized control and classification tasks because of human brains' brilliant mechanisms. However, emotion detection such as differentiating genuine anger and acted anger, sometimes, is still difficult for human beings as they have learned to hide their emotions in social environments. Therefore, in this paper, neural networks with physiological signals, pupillary responses, as input patterns are design to assist people to distinguish genuine and posed anger and a network pruning technique is implemented to improve these models. The results show that a standard neural network can provide only 66.25% test accuracy, while a recurrent neural network (RNN) can provide 87.82% accuracy. However, pruning technique can improve standard neural network to 86.25% test accuracy, while the same pruning technique cannot improve RNN models in this paper.

Keywords: standard neural network, recurrent neural network, weight pruning, anger detection, pupil parameters

1 Introduction

1.1 Background

Human beings are good at expressing different emotions, and human brains are capable to tell different emotions. However, people, sometimes, mask their emotions in social events for various reasons, and people may face difficulties to tell the veracity of emotions [1]. Therefore, there are many studies related to detecting the veracity of human emotions in machine learning and neural network fields. Physiological signals such as pupil parameters are widely used in these studies as theses parameters are independent of conscious human control and have measurement results are more objective [2].

This paper is inspired by the research of [1] to study how well neural networks can differentiate genuine and posed anger given pupillary responses as input. It is also an extension of previous paper [3] to compare the prediction accuracy of standard neural network and recurrent neural network. In addition, pruning technique is applied to all of these models to compare how this technique can improve different type of models.

1.2 Motivation

Pupils are one of the most important body tissues, and the reaction of pupils are mostly at unconscious level with little conscious judgment [4]. My experiment attempts to show that pupil parameters can be valid input features for emotion detection. In addition, this paper tries to select a model from various a set of standard neural network models and recurrent neural network models with different complexity as it significantly affects prediction accuracy. A model should be complex enough to capture input features, while it should not be too complex to memorize particular input patterns. Therefore, the idea of network pruning also comes up to generalize the model and make it more robust to testing data.

1.3 Research Question

In this paper, two type of neural network models (i.e. standard neural network and recurrent neural network) are design to predict whether an anger expression is posed or not, given pupillary responses as input data. The prediction accuracy of these two type of models are compared to show the effectiveness of these models in this scenario. In addition, pruning technique are applied to both models to show how this technique can improve standard neural network and recurrent neural network.

2 Method

2.1 Tools and Data Selections

Python and PyTorch are used to implement the neural network models designed in this paper as it provides some high-level features such as tensor computation via graphics process units, automatic differentiation system etc.

2.1.1 Data Selections

Both standard neural network models and recurrent neural network models use pupillary responses from "Anger" dataset collecting from movies and documentary videos as input data[5-6]. Based on the characters of different network structure, standard neural network models use statistic pupillary responses as input data, while recurrent neural network models use time series data of pupillary responses coming from the same video streams.

Input data for standard neural network models contain 400 samples in total with 200 samples labeled as "genuine" and 200 samples labeled as "posed". Each sample has 9 columns (i.e. 9 attributes) shown in table 1. "Index" refers to the number of samples for each single videos (i.e there are 20 samples for each video). "Video" is the name of video. "Mean", "Std", "Diff1", "Diff2", "PCAd1", and "PCAd2" refer to the statistic data of pupil parameters. They store the mean, standard deviation, minimum, maximum, mean of the absolute values of the first differences of the processed signals, mean of the absolute values of the second differences of the processed signals respectively.

Index	Video	Mean	Std	Diff1	Diff2	PCAd1	PCAd2	Label

Table 1. Column Headers of "Anger" Dataset [5]

Input data from recurrent network models are date in time series coming from three separated Excel files. The first file, "MeanPD_Anger", contains 3 sheets. The first sheet contains 20 columns, and each columns represents the mean values of tested subjects' left pupil diameters of one video. The second sheet contain the same information except that the data are related to tested subjects' right pupil diameters. The third sheet combines the first two sheets and calculates the mean values of tested subjects' pupil diameters. It also adds two extra columns storing the mean values of pupil diameters for all "Genuine" and "Posed" anger of each video respectively. The second file, "PDleft", contains 20 sheets, and each sheet stores the data related to tested subjects' left diameters in time series of one single video. Each column represents one tested subject and there are twenty tested subjects in each sheet. The third file, "PDright", contains same information as the second one except that all the data are related to tested subjects' right diameters. In total, there are 780 samples with 392 samples labeled as "genuine" and 388 samples labeled as "posed".

2.1.2 Data Preprocessing

In terms of input data for standard neural network, the last column "Label" is used as a target label to show whether the anger in the video is posed or not. Before training, string values in this column will be changed to integers (i.e. "genuine" as 1 and "posed" as 0). In addition, a popular physiological signal in emotion detection, pupil reaction, will be used in this study. "Index" and "Video" columns only represent data and video order which have no effect on tested subjects' pupil reactions. Therefore, these two columns will not be included as input features, and all other columns are treated as input features to feed into the standard neural networks as they are all related to tested subjects' pupil reactions. Moreover, all these six columns have been normalized under z-score; thus, the normalization step can be skipped in this case.

In terms of input data for recurrent neural network, "labels" are not directly provided, but data sheet name indicates the label (i.e. "T" refers to "Genuine" and "F" refers to "Posed). Therefore, an label column is added based on it data sheet name. Different video has different time series length, and the shortest time step 60 is taken for all videos to make the input tensor of different videos even with each other. In addition, to make the time series data more smooth, all zero data are replaced with median values of its time series. Finally, all the time series data will be normalized under z-score.

Both dataset will be split into two parts i.e 80% data as training set and 20% data as testing set. Data should be randomly drawn as training data or testing data, and training set and testing set should have the same distribution. Therefore, for standard neural network input data, data with "genuine" label should be shuffled and drawn 160 data as training data and 40 data as testing data. The same procedure should be applied to data with "posed" label as well. Then, "genuine" training data should be combined with "posed" training data, and the entire training set should be fully shuffled. This procedure should be applied to "genuine" testing data and "posed" testing data as well. In addition, the same operation should be applied to recurrent neural network input data as well except that there should be 624 training samples and 156 testing samples.

2.2 Neural Network Models

There are two type of models designed in this paper i.e. standard three layer neural network models and recurrent neural network models, specifically, long short-term model (LSTM). The aim of this paper is to compare the effectiveness of these two type of models and different model sizes will influence the prediction accuracy; therefore, all other hyperparameters except model type and model size are held. In all models, Adam is used for optimization; binary cross entropy loss is used to calculate training loss in each iteration; learning rate is set to be 0.001; epoch is set to be 4000. The training results and testing results are evaluated through accuracy score (i.e. No. of correct classification/ total No. of samples).

2.2.1 Three Layer Neural Network Models

ThreeLayerNetwork class in model.py of network_pruning project defines a standard three layer neural network (i.e. one input layer, two hidden layer and one output layer), all of these layers are fully connected. The input layer takes six input feature discussed in section 2.1. As this is a binary classification problem, there is only one neuron in the output layer. Four different standard neuron networks models with different number of neurons in two hidden layers are designed and trained (shown in table2), and their prediction accuracy will be tested. The activation function for the two hidden layers is tanh function as tanh function usually exhibits better properties for training than sigmoid function [7], and the activate function for output layer is sigmoid function as this is a binary classification problem.

Model type	No. of Neurons in hidden layer1	No. of Neurons in hidden layer2
super small model	5	3
small model	20	10
medium model	100	50
large model	200	100

Table 2. Standard Neural Network Model with Different Size

2.2.2 Recurrent Neural Network Models

LSTM class in model.py of improve_network project defines a recurrent neural network model, specifically a long short-term model (LSTM). It takes time series data discussed in section 2.1 as input and the first hidden layer is a standard LSTM layer. Afterwards, this LSTM layer is connected to a fully connected layer and an output layer. To make these two type of models more comparable, the number of neurons in each layer are the same as the number of neurons in three layer standard neural network models (shown in table3), and their prediction accuracy will be tested as well. In addition, the activation function for fully connected network in LSTM is also tanh function and activation function for output layer is also sigmoid function which is same as the standard three layer neural network model.

Model type	No. of Neurons in LSTM layer	No. of Neurons in fully connected layer
super lstm small model	5	3
small lstm model	20	10
medium lstm model	100	50
large lstm model	200	100

Table 3. LSTM with Different Size

2.3 Network Pruning

Network pruning technique is applied to models discussed in section 2.2, and there are two pruning strategies, i.e neuron pruning and weight pruning. Neuron pruning strategy prunes neurons with similar functionality in hidden layers. Neuron functionality is identified by the distinctiveness property of hidden neuron through distinctiveness angular measure described by Gedeon and Harris [8]. The distinctiveness angular is calculated through each pair of column vectors with the following formula:

$$\text{angle}\ (v1, v2) \ = \ \arccos\left(\frac{v1 \cdot v2}{||v1|| \ \cdot ||v2||}\right)$$

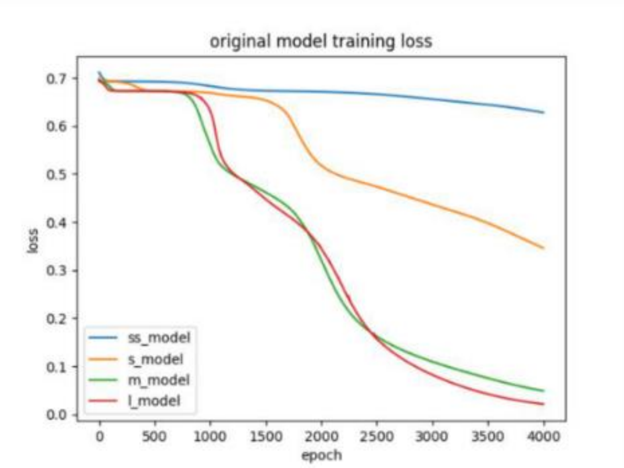
Usually, if the angle is less than 15, two vectors will be regarded as similar vectors and one neuron should be removed. If the angle is greater than 165, two vectors will be regarded as complementary vectors and both of the neurons should be removed. While weight pruning strategy prunes weights in hidden layers that only have tiny influence on the neurons. In this paper, the second strategy is used to improve both standard neural network model and recurrent neural network model. L2-norm of the weight is used to measure the influence of each weight. Moreover, weight pruning will only apply to medium and large size of both models as the other models are too small to prune certain amount of weights and the pruning amount varies from 10% to 60% of total weights.

3 Result

The follow plots and tables compare training and testing accuracy of different size of standard three layer neural network models and LSTM models. Theses plots and tables also show the testing accuracy of different size of these two type of models with different weight pruning. It is obvious that medium and large LSTM models performance much better than medium and large standard neural network models before pruning. Medium and large standard neural network models with proper weight pruning can achieve similar test accuracy, but weight pruning does not significantly improve the performance of LSTM models.

3.1 Results of Standard Neural Network Models

Figure1 and figure2 shows the training loss and training accuracy of four standard neural network with different size before pruning. On average, the larger size of the model, the higher training accuracy and the less training loss it can achieve. However, the testing accuracy of larger model (0.6625) is much smaller than its training accuracy (1.0) shown in table4.



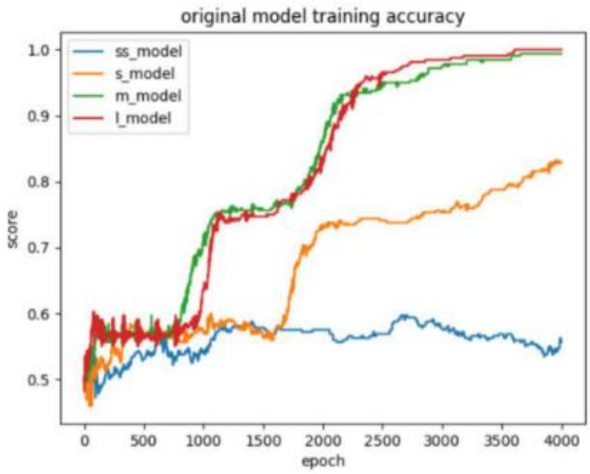
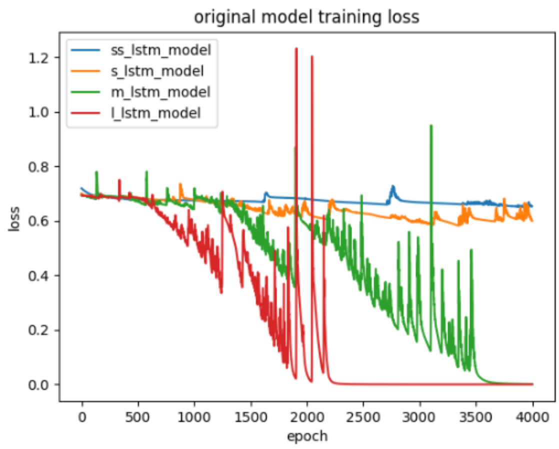


Figure1. Training Loss of Standard Neural Network Models[3] **Figure2.** Training Accuracy of Standard Neural Network Models[3]

3.2 Results of LSTM models

Figure3 and figure4 shows the training loss and training accuracy of four diffent size LSTM models before pruning. It is obvious that, super small and small LSTM models cannot achieve either acceptable training accuracy or testing accuracy (only around 55%) shown in table4. The large LSTM model can achieve 87.82% testing accuracy after training, while large standard neural network can only achieve 66.25% testing accuracy even it has 100% training accuracy. It is also worth to notice that the training accuracy of LSTM models are much more volatile than standard neural network models'.



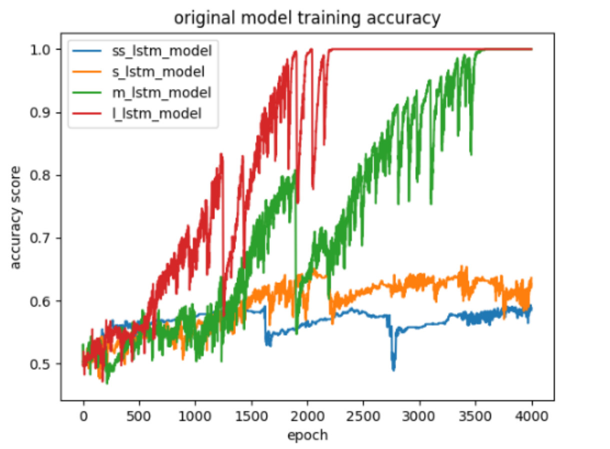


Figure3. Training Loss of LSTM **Figure4.** Training Accuracy of LSTM

Model type	training accuracy	testing accuracy
super small model	0.55623	0.5375
super lstm small model	0.5625	0.5369
small model	0.8281	0.6375
small lstm model	0.6273	0.5625

medium model	0.9938	0.6125
medium lstm model	1.0	0.7259
large model	1.0	0.6625
large lstm model	1.0	0.8782

Table 4. training and testing accuracy of different models

3.3 Results of pruned models

Table5 shows testing accuracy of two type of models with different pruning percentages. On average, for both type of models, the more weight pruned, the less testing accuracy is. For standard neural network models, test accuracy improves significantly when pruning amount is less than 40% (with at least 76.25% testing accuracy). However, weight pruning does not significantly affect LSTM models (87.82% before pruning and 86.06% after pruning for LSTM models).

Model type	pruning percentage	test accuracy
medium model	10%	0.8375
medium lstm model	10%	0.8253
medium model	20%	0.7875
medium lstm model	20%	0.7596
medium model	30%	0.8375
medium lstm model	30%	0.7708
medium model	40%	0.8
medium lstm model	40%	0.7131
medium model	50%	0.5375
medium lstm model	50%	0.6843
medium model	60%	0.475
medium lstm model	60%	0.5753
large model	10%	0.825
large lstm model	10%	0.8606
large model	20%	0.8375
large lstm model	20%	0.8108
large model	30%	0.875
large lstm model	30%	0.7917
large model	40%	0.8625
large lstm model	40%	0.7564

large model	50%	0.7625
large lstm model	50%	0.6683
large model	60%	0.475
large lstm model	60%	0.5224

Table 5. testing accuracy of different pruned models

4 Discussion

Intuitively, large models usually generate better results than small models as the complexity of the architecture helps to capture more input patterns. This paper can prove this opinion as LSTM models perform overall better than standard neural network models and LSTM models have four times parameter than standard neural network models when having same number of hidden neurons. Section3 shows that neither small size standard neural network models nor LSTM models can achieve high training and testing accuracy. Probably because these models are too simple to capture high level features which resulting in underfitting. While large size LSTM models perform better than large size standard neural network models. This may because LSTM models are more complex than standard neural network models which can capture more input patterns. In addition, LSTM models have four times parameters than standard neural network models; therefore, these models are more sensitive to input data and phenomenon aggravates as the size of model increases. In this experiment, both standard neural network models and LSTM models face overfitting issue as the training accuracy for large models are 1 while the testing accuracy are only 66.25% and 87.82% respectively for standard neural network model and LSTM model. Because there are limited data (400 samples for standard neural network models and 780 samples for LSTM models) used for training and the models can simple memorize some patterns.

Many researches such as “The Lottery Ticket Hypothesis” [9] find that network pruning is an efficient techniques to generalize model and relive overfitting issue. However, this experiment finds that network pruning only works for standard neural network models. This may because LSTM models are not over complicated and reducing model complexity cannot help to improve models. However, the number of training data is not enough to properly train a model with such amount of parameters. Moreover, the experiment also finds that the testing accuracy will drop significantly when the pruning amount is large (i.e. over 40%) as some weights which have significant influence on the network have been pruned.

5 Conclusion

In conclusion, feeding pupillary responses data into neural network models can provide a relatively good prediction and LSTM models perform better than standard neural network. However, this is not as good as predecessors’ prediction result (95%) [1]. It still can show that pupillary responses can be a good indicator of genuine anger and this indicator can be applied to explore some other human emotions in future work. In addition, network pruning can generalize model and improve prediction accuracy to some extend, but sometimes, enough training data is imperative.

6 Future Work

In this paper, only one pruning technique, i.e weight pruning has been implemented and some other network pruning techniques such as neuron pruning can also be implemented. It may achieve better result than weight pruning as it focus on pruning similar and complementary neurons which are more target on neurons with similar functionality. In addition, this paper focuses only on comparing the effectiveness of standard neural network and recurrent neural network and how network pruning can improve these networks. Therefore, the experiment changes only the network type and neuron numbers and holds all other factors such as learning rate, loss function, optimizer etc. However, all these factors may work together to affect the model prediction accuracy and further research can be done to explore how other hyperparameters in a model will affect prediction result.

7 References

[1] Chen, L., Gedeon, T.D., Hossain, M.Z., Caldwell, S.: Are you really angry? Detecting emotion veracity as a proposed tool for interaction. In Proceedings of the 29th Australian Conference on Human-Computer Interaction (OzCHI '17), pp. 412--416 (2017)

[2] Zhihong Zebg, Jilin Tu, Ming Liu, Tong Zhang, Nicholas Rizzolo, Zhengqiu Zhang, Tomas S. Huang, Dan Roth, and Stephen Levinson. 2004. Bimodal HCI-related affect recognition. In Proceedings of the 6th international conference on Multimodal interfaces (ICMI 04). ACM, New York, NY, USA, 137-143. DOI:

http://dx.doi.org/10.1145/1027933.1027958

[3] E. Gao. Improve Neural Networks through Network Pruning. Australian National University (2021) .

[4] Alaoui-Ismaïlia, O., Robin, O., Rada, H., Dittmar, A., Vernet-Maurya, E.: Basic Emotions Evoked by

Odorants: Comparison Between Autonomic Responses and Self-Evaluation. Physiology & Behavior. Volume 62,

Issue 4, 713--720 (1997). https://doi.org/10.1016/S0031-9384(97)90016-0

[5] Anger.xlsx (2021). DOI: http://cs.anu.edu.au/~tom/datasets/anger.zip

[6] Anger_v2 (2021) DOI: http://cs.anu.edu.au/~tom/datasets/anger_v2.zip

[7] Kalman, Barry L., and Stan C. Kwasny. "Why tanh: choosing a sigmoidal function." *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*. Vol. 4. IEEE, 1992.

[8] Gedeon, T. D., and D. Harris. "Network reduction techniques." Proceedings International Conference on Neural Networks Methodologies and Applications. Vol. 1. 1991.

[9] Frankle, Jonathan, and Michael Carbin. "The lottery ticket hypothesis: Finding sparse, trainable neural networks." *arXiv preprint arXiv:1803.03635* (2018).