Detecting genuine and posed angry emotions by pupil parameters with shallow neural network and recurrent neural network

Xiaoshi Lu

Research School of Computer Science, Australian National University, Canberra, Australia
U6511909@anu.edu.au

Abstract. Detecting human emotions is challenging. Inspired by biological researches, many professionals recognize pupil parameters as useful tools to detect complex emotions. I use a shallow neural network with extracted pupil parameters as input patterns to judge if the participants are exposed to genuine or posed angry emotions. To improve this neural network, I apply Characteristic Input Pattern technology. I also build a recurrent neural network with raw pupil parameters as input patterns to do the same work. At last, I compare these technologies. The results prove that pupil parameters are valid in detecting angry emotion. Besides, both shallow neural network and recurrent neural network can give good predictions. Characteristic Input Pattern can help the shallow neural network, but it costs more training resources. This means that we could further explore pupil parameters or other physiological data by shallow neural networks or recurrent neural networks to detect more emotions in future.

Keywords: angry emotion, pupil parameters, neural network, recurrent neural network

1 Introduction

This part will introduce the related background information, the motivation why I choose this topic and what specific questions I will discuss in the following contents.

1.1 Background

Human emotions are one of the most popular topics in the machine learning field, not only because companies can gain more profits by considering human emotions when they develop new products, but also because the understanding of human emotions can help us investigate more deeply in anthropology. Human emotions are complex, but from facial expressions, we could recognize some of them in our daily life. For example, the happiness is with smiles; the sadness is with tears, and the furiousness is with yelling [1]. However, since there are other interferences such as gestures, languages, and contexts, it is difficult to always give correct judgements. In some situations, people will also deliberately express some posed emotions to achieve their goal. As a result, even human brains may make mistakes when try to detect the genuine and posed emotions of other people [2].

To solve this question, some professionals propose that we could use physiological data to assist emotion detecting [3]. Since pupils are critical parts for emotion detecting and expressing, many researchers wish to use pupil parameters collected from subjects who are exposed to emotional signals and judge what the emotions are. Based on the research of [2], this method is helpful both for angry detecting and smile detecting.

1.2 Motivation

I used to participate in the data collecting process in similar research. It also aimed to collect pupil parameters so that researchers can judge if one specific angry emotion signal is genuine. However, as a tested subject, I am not sure if the pupil parameters are as useful as we thought since many other factors were affecting my reactions and feelings during the test. Although many professionals certify that pupil parameters will not be influenced severely by outside factors and are more reliable than human being's conscious judgement, I still want to prove it by myself [4].

1.3 Research Question

I am given two datasets. The first one is "anger" with extracted pupil parameters. The second one is "anger_v2" with raw pupil parameters in timeseries. I build a shallow neural network and train a binary classification model with "anger" to see if the model can differentiate genuine angry emotions from posed ones. Then I insert Characteristic Input Pattern technology to see if it will help the shallow neural network classification model. On the other hand, I build a recurrent neural network with long short-term memory (LSTM) and train a binary classification model with "anger_v2" to do the same work. If I could get a testing accuracy which is greater than 50%, it will prove that pupil parameters can help detect genuine and posed angry emotions.

2 Method

This part will discuss the tools, data selections and the methods that I use in this paper with details.

2.1 Tools and Data Selections

I use Python and PyTorch to develop my shallow neural network and recurrent neural network. This is because PyTorch is a mature neural network development tool. It is "Production Ready, Distributed Training, Robust Ecosystem and Cloud Support" [5].

As for data selections, the first dataset "anger" consists of 400 samples and each sample contains 9 elements. The headers of these samples are shown in Table 1.

Table 1. The headers of samples in dataset "anger".

Index	Video	Mean	Std	Diff1	Diff2	PCAd1	PCAd2	Label

"Index" stores the counters of samples for each angry signal video. The counter will wrap from 0 when it reaches 20. In total there are 20 (videos) * 20 (samples) = 400 (samples) in the whole dataset. "Video" stores the name of angry signal video for each sample. Since the "Index" and the "Video" do not affect subjects' pupil parameters, I drop these two columns of the data during the training of the shallow neural network. "Mean", "Std", "Diff1", "Diff2", "PCAd1", and "PCAd2" store the mean values, standard deviation values, minimum, maximum, means of the absolute values of the first and second differences of the processed signals respectively [6]. These six columns have already been normalized to 0 mean and 1 standard deviation. They are all pupil parameters and are treated as input features of the shallow neural network. Meanwhile, the "Label" infers whether one specific angry signal video is "Genuine" or "Posed", so this column is treated as the outputs or targets.

The second dataset "anger_v2" is much more complex, containing raw data in timeseries. There are 3 Excel files.

The first Excel file "MeanPD_Angry" consists of 3 sheets. The first sheet has 20 columns, and each column stores the means of subjects' left pupil diameters for a specific angry signal video. The second sheet is almost the same as the first one, but it stores the means of subjects' right pupil diameters for 20 angry signal videos in 20 columns. The third sheet combines the data in the first and second sheet, calculating the means of subjects' pupil diameters for 20 angry signal videos in 20 columns. The third sheet also introduces another 2 columns, respectively storing the means of pupil diameters for all "Genuine" angry signal videos and all "Posed" angry signal videos. All the columns in "MeanPD_Angry" are data in timeseries.

The second Excel "PDleft" consists of 20 sheets named from "T1" (the first genuine angry signal video) and "F1" (the first posed angry signal video) to "T10" (the 10th genuine angry signal video) and "F10" (the 10th posed angry signal video). Each sheet records the data related to one specific angry signal video. In each sheet, there are 20 columns. Each column stores one subject's left pupil diameters in timeseries for the angry signal video. Totally, there are 20 (angry signal videos / sheets) * 20 (participants / columns) = 400 samples.

The third Excel "PDright" is almost the same as "PDleft", but it stores subjects' right pupil diameters in timeseries for 20 angry signal videos. Totally, there are also 20 (angry signal videos / sheets) * 20 (participants / columns) = 400 samples.

I mainly use the 800 samples in "PDleft" and "PDright" to train the recurrent neural network model because they are raw data, and recurrent neural networks have more advantages when dealing with raw data. Since the shortest angry signal video "F1" has 60 timesteps, all the timeseries samples are extracted their first 60 timestep data to form a meaningful input tensor.

2.2 Shallow Neural Network

In the first stage, based on the first version dataset "anger", I define a shallow neural network with one input layer, one hidden layer, and one output layer. The training losses and the testing accuracies will be expressed in plot figures and 2*2 confusion matrices.

Since the data in "anger" has already been normalized, there is no need to do another normalization. I shuffle the whole dataset. dividing it into a training dataset (80% of the whole dataset) and a testing dataset (20% of the whole dataset).

The input layer contains 6 input neurons, standing for 6 input features: "Mean", "Std", "Diff1", "Diff2", "PCAd1", and "PCAd2".

The hidden layer contains 10 neurons, with the sigmoid activation function. Sigmoid activation function sometimes will suffer from the "vanishing gradients" [7]. However, this is a shallow neural network for binary classification, so "vanishing gradients" is not serious and will not affect the predictions deeply.

The output layer contains 2 neurons. As a result, the output for each input pattern will be a 2-dimension tensor, and I will use the index of the maximum element in the tensor to stand for the class number. The label of "Genuine" is 0, and the label of "Posed" is 1.

By default, the learning rate is 0.01; training epoch number is 1000; loss function is Cross-Entropy Loss; optimizer is Adam. All these hyperparameters can be set in the main function.

2.3 Characteristic Input Pattern

Gedeon and Turner used to propose a concept called Characteristic Input Pattern which could help improve classification neural networks [8]. The main method is also explained in [9]. One of the key steps is to define the Characteristic Input Pattern for each class, and usually, the mean of input vectors for one specific class is applied. Then use characteristic input patterns to find significant input units. With this information, rules can be generated to assist classifying. [9]

In this research, I also use the mean of training data input patterns from "Genuine" class in the first version dataset "anger" as the Characteristic Input Pattern.

To analyze how the shallow neural network to make decisions, I will set one of the features in the Characteristic Input Pattern to 0. Pass the changed input pattern to the shallow neural network model and get the prediction. Then I gradually increase the value of this feature. After each increment, I pass the whole input pattern to the shallow neural network model and get the prediction until I find a turning point.

The above steps will be reiterated for all the features in the Characteristic Input Pattern.

For example, if the Characteristic Input Pattern for "Genuine" class is [0.8861, 0.1025, 0.0082, 0.2050, 0.0303, 0.1183]. First, I will set the first feature 0.8861 to 0 and keep other features unchanged. Pass the [0, 0.1025, 0.0082, 0.2050, 0.0303, 0.1183] to the neural network model and get the prediction. If the output is 1 ("Posed"), which means a false prediction, I will increase the feature value by 0.02, pass the new input pattern through the neural network model and get a new prediction. Keep iterating until it reaches 0.86 and give a different output 0 ("Genuine"). It means I find the turning point and get one rule: "The first feature should be greater or equal to 0.86 for 'Genuine' class." The upper bound of each feature value is 1 since the data is normalized. Reiterate these steps for all the features of Characteristic Input Pattern. The features whose change of their values can affect the prediction are significant input units.

2.4 Recurrent Neural Network with LSTM

Recurrent neural networks (RNN) [10] are prevailing deep learning methods nowadays. RNNs can deal with variable-length sequence inputs and extract high-level feature representations automatically rather than manually. Typically, the inputs will be expressed by a matrix (sequence dimension * the feature vector dimension). Long short-term memory (LSTM) is a variant of RNN [11]. It is more robust when facing with vanishing gradient or exploding gradient problems [12].

In the second stage, based on the second version dataset "anger_v2", I define a recurrent neural network with LSTM with one input layer, one hidden layer, and one output layer. The training losses and the testing accuracies will be expressed in plot figures and 2*2 confusion matrices.

The raw data needs to be pre-processed.

First, there are some samples with nan value, which means the subject for this angry signal video was absent. I drop these samples because they are useless. Second, the dataset needs to be normalized. Here I use the Z-score method to get data with 0 mean and 1 standard derivation. One point is that I do not directly use the mean of the whole dataset during normalization because many samples are containing 0 values (eye blinks) in timeseries. To get rid of the affections of these 0 values, I use the median to replace these 0 values and then apply Z-score method.

I also shuffle the whole dataset. dividing it into a training dataset (80% of the whole dataset) and a testing dataset (20% of the whole dataset).

The training uses mini-batch technology. The input training data is grouped in many tensors with dimensions of batch* timestep*input size. As mentioned in Section 2.1 Tools and Data Selections, the timestep is 60 (Each sample has been extracted their first 60 timesteps) and the input size is 1 (Each sample is one column of data).

The hidden layer contains 30 neurons, half of the timestep 60.

The output layer contains 2 neurons. As a result, the output for each input pattern will be a 2-dimension tensor, and I will use the index of the maximum element in the tensor to stand for the class number. The label of "Genuine" is 1, and the label of "Posed" is 0, which is different from the labels in the shallow neural network.

By default, the learning rate is 0.001; training epoch number is 200; loss function is Cross-Entropy Loss; optimizer is Adam; batch size is 10. All these hyperparameters can be set in the main function.

3 Result

This part will discuss the results of shallow neural network (without and with the Characteristic Input Pattern) and recurrent neural network. Then I will combine the results.

3.1 Shallow Neural Network

After 3000 epochs training, the training loss reduces from 0.7911 to less than 0.5381, while the testing accuracy increases from 50% to nearly 75%.

The plots of training losses and testing accuracies are shown in Figure 1.

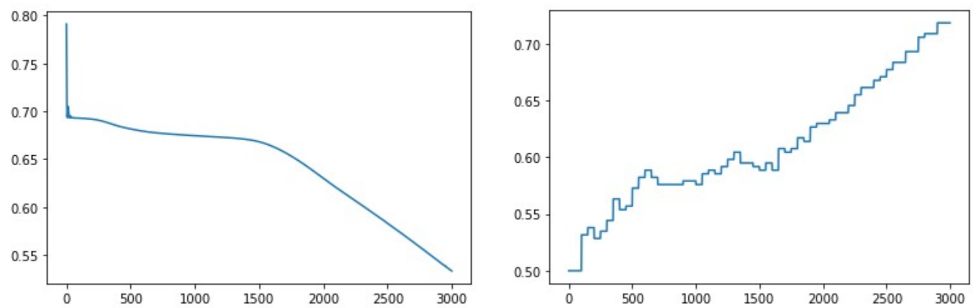

Fig. 1. The training losses (left) and testing accuracies (right) of shallow neural network.

3.2 Characteristic Input Pattern

The quality of generated rules depends on the chosen model. If the model has been trained well, it is challenging to find rules because the change of a single feature in the Characteristic Input Pattern cannot affect the outputs obviously. On the other hand, if the model has not been trained enough, it is also difficult to find the rules because the new model prefers to give different inputs to the same outputs. One of the rules I generated is: "Std>0.075 AND Diff2>0.08 AND PCAd2>0.075" (For "Genuine" class).

To get used of Characteristic Input Pattern technology, I first compare each input pattern with the rule generated from "Genuine" Characteristic Input Pattern. If the input pattern obeys the rule, then directly classify this input pattern to "Genuine" class. Otherwise, pass it to the shallow neural network to get the prediction. In this way, the training loss reduces to around 0.4670, while the testing accuracy increases to more than 75%.

The plots of training losses and testing accuracies are shown in Figure 2.

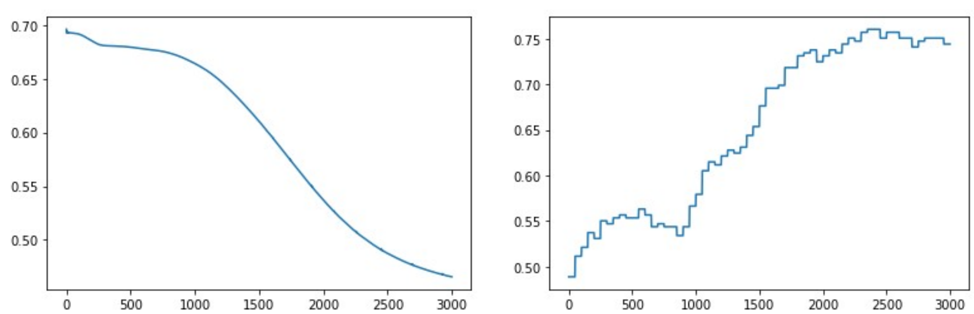

Fig. 2. The training losses (left) and testing accuracies (right) of shallow neural network with Characteristic Input Pattern.

3.3 Recurrent Neural Network with LSTM

After more than 1000 epochs training, the training loss reduces from 0.6948 to less than 0.175, while the testing accuracy increases from 48% to more than 70%. Since the testing dataset does not always have the same number of samples for two classes and the newly initialized model prefers to give an all 1s or 0s prediction, it is likely to have a less than 50% testing accuracy in the first a few epochs. After some training, the testing accuracy can increase. The plots of training losses and testing accuracies in the initial stage and the last stage are shown in Figure 3 and Figure 4.

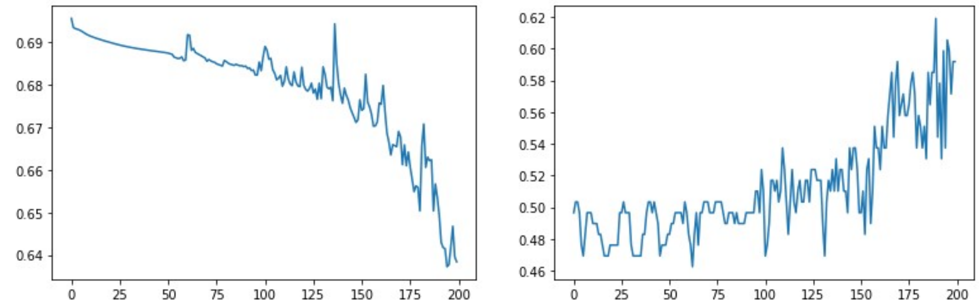

Fig. 3. The training losses (left) and testing accuracies (right) of RNN with LSTM in the initial stage.

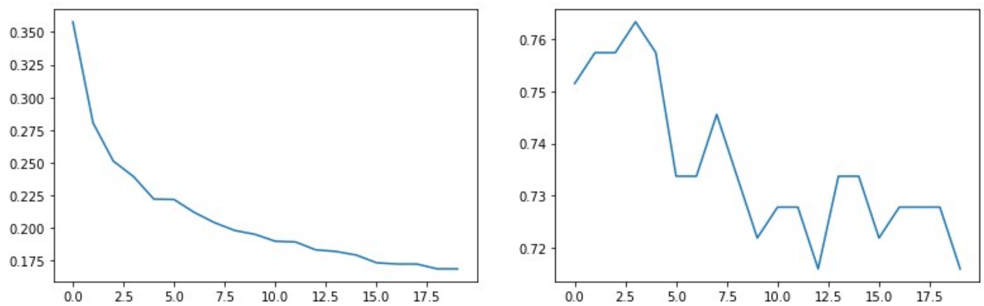

Fig. 4. The training losses (left) and testing accuracies (right) of RNN with LSTM in the last stage.

In the final training stage, the training loss is still decreasing but the testing accuracy begins dropping. This implies a probability of overfitting, so I stop the training.

Another point is that during the training of recurrent neural network in Anaconda Spyder, if the Console has already worked for a long period with innumerable calculations before the training and if the epoch number is too large, it is likely to break off and throw a run time error: "RuntimeError: [enforce fail at ..\c10\core\CPUAllocator.cpp:72] data. DefaultCPUAllocator: not enough memory: you tried to allocate 1144800 bytes. Buy new RAM!" This is because the Console is too tired, and the allocated cache is full. To solve this problem, the only thing needed is to close the current Console and open a new one.

3.4 Combination

As the results are shown above, both shallow neural network and recurrent neural network can train a model, detecting genuine and posed angry emotions by pupil parameters. Adding Characteristic Input Pattern technology to the shallow neural network will also improve the testing accuracy a little, but it trades off with calculation performance.

4 Discussion

This part talks about the answer to the research question, the comparison of techniques, the limitation of researches, and the direction of future study.

4.1 Answer

Although all the testing accuracies (shallow neural network without/with Characteristic Input Pattern and recurrent neural network with LSTM) are not as good as predecessors in emotion veracity detection (95%) [2] and in explaining students' grade with characteristic input patterns (94%) [8], all the neural networks can use pupil parameters as input patterns to give a relatively good prediction accuracy of angry emotion (at least 70%) which is higher than the random judgement accuracy (50%).

It proves that pupil parameters can help a lot in detecting genuine and posed angry emotions with shallow neural networks and recurrent neural networks. Put it another way, pupil parameters are good reflections of human being emotional signals. In the future, it encourages us to explore other emotions with pupil parameters.

4.2 Comparison

The shallow neural network has its advantages. It is easy to build. The training is fast, and the prediction is relatively good. However, there are some disadvantages. The shallow neural network is skilled in processing carefully extracted data, which demands large amounts of pre-processing work.

The Characteristic Input Pattern can slightly increase the testing accuracy of the shallow neural network model, but the calculation performance will be affected heavily since it needs to do additional work for each input pattern. Besides, it does not always work well for all the models with different training levels. Since the second version raw dataset “anger_v2” is much larger than the first version dataset “anger”, it is too expensive to apply Characteristic Input Pattern technology in the recurrent neural network.

The recurrent neural network with LSTM can professionally process raw dataset. It is not difficult to build with Pytorch, and the prediction is relatively good. However, it takes much more time and resources to train a usable model.

In total, a shallow neural network is suitable for an extracted dataset and quick training. Characteristic Input Pattern could be applied when the dataset is small so that it would not affect the calculation performance too much. A recurrent neural network with LSTM is appropriate when a large raw dataset, a good CPU or GPU, and enough training time are offered.

4.3 Limitation

Because of the time limitation, I do not keep training until there is a convergence. It means I do not find the optimal solution, but just find a relatively acceptable solution with local minimum.

4.4 Future Study

For future study, it is worthwhile to use pupil parameters and detect other kinds of emotions using shallow neural networks or recurrent neural networks.

Besides, we could also collect other physiological signals such as the pulse rate of subjects who are exposed to angry signal videos. Combine different kinds of physiological signals including pupil parameters and train a new model which detects genuine and posed angry emotions by using neural network technologies mentioned above.

Another direction is that we could use generative modelling to produce some artificial angry signal videos. Pass the data related to these artificial videos and real videos through the existing neural network and build a new generative adversarial network. Analyze the new network, we may find out some common and unconscious features of human angry emotions.

References

1. Ekman, P.: An argument for basic emotions. Cognition and Emotion. Volume 6, Issue 3-4, pp. 169--200 (1992)
2. Chen, L., Gedeon, T., Hossain, 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)
3. Domínguez-Jiménez, J.A., Campo-Landines, K.C., Martínez-Santos, J.C., Delahoz, E.J., Contreras-Ortiz, S.H.: A machine learning model for emotion recognition from physiological signals. Biomedical Signal Processing and Control. Volume 55 (2020)
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)
5. PyTorch, https://pytorch.org/
6. Hossain, Z., Gedeon, T.: Classifying Posed and Real Smiles from Observers’ Peripheral Physiology. 11th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth '17), EAI Conference Series, Barcelona, Spain, 1-4. Retrieved from http://bit.ly/2gkllCZ
7. Sharma V, A.: Understanding Activation Functions in Neural Networks. The Theory of Everything. https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0
8. Gedeon, T., Turner, S.: Explaining student grades predicted by a neural network. In Neural Networks, 1993. IJCNN'93-Nagoya. Proceedings of 1993 International Joint Conference. Volume 1, pp. 609—612. IEEE Press (1993)
9. Turner, H.S., Gedeon, T.: Extracting Meaning from Neural Networks. http://cs.anu.edu.au/~tom/papers/ExtractingMeaning.pdf
10. Elman, J.L.: Distributed representations, simple recurrent networks, and grammatical structure. Mach. Learn. Volume 7, Issue 2–3, pp. 195-225 (1991)
11. Xu, J., Chen, D., Qiu, X., Huang, X.: Cached long short-term memory neural networks for document-level sentiment classification. arXiv preprint arXiv:1610.04989.
12. Liu, F., Zheng, J., Zheng, L. Chen, C.: Combining attention-based bidirectional gated recurrent neural network and two-dimensional convolutional neural network for document-level sentiment classification. Neurocomputing. Volume 371, pp. 39-50 (2020)