Feature engineering and multimodal fusion of neural network training

Yizhe Xin
Research School of Computing
The Australian National University
Canberra ACT 2601
u7155031@anu.edu.au

Abstract. It is a very meaningful and challenging task to judge the patient's depression tendency through the patient's physiological response data. Thanks to the rapid development of neural networks, the process of mathematical modeling has become more convenient and effective. In this work, I tried to train the neural network model using the three characteristics of galvanic skin response, skin temperature and pupil dilation.To classify the degree of depression in patients. **In my work, I used xgboost for feature selection of the original data, and compared to assist2, I used a deeper and wider neural network to complete the work. From the results, it can be seen that my updated method has achieved Better results.**

Keywords: Depression Detection, Neural Networks, Feature selection， Multimodal Fusion

1 Introduction

Thanks to the rapid development of neural network in recent years, it has exceeded the accuracy of human in some classification tasks, such as identifying the degree of individual depression in video [1]. However, classification using neural networks is usually based on a single type of input. For example, distinguish a cat from a dog based on a picture or sound. It is not difficult to find that in real life, there are many kinds of characteristics describing the same attributes of objects. Therefore, it is generally believed that the integration of multimodal information in the classification task is bound to help our neural network to complete the classification task. This method is technically called multimodal fusion. Now many scholars are committed to the research in this field, and have achieved good results[2]

As for the detection of emotional response to depression videos, we have three physiological signals, namely, skin electrical response, skin temperature and pupil dilation, provided by Xuanying Zhu et al. In their work, they think that not all the features are conducive to the training of neural network. So they use GA algorithm to choose the best feature combination. This can help the neural network to accurately predict the degree of depression. They did rigorous experiments to test their emotions.

Because the real information is very complex and rich, sometimes the information we collect in the real world is redundant for downstream tasks. Therefore, we need to carry out feature engineering before neural network training, that is, feature analysis and feature selection. Thanks to the development of xgboost[3] series of algorithms, we can use xgboost to train a simple model of the existing data, so as to carry out feature selection. When training neural network, we only use the selected features. In addition, the neural network I used in Assignment 1 is too shallow, so I used deeper and wider network to model in this work. Based on this model, the comparison between algorithms is reasonable.

2 Data

Because I didn't give a detailed introduction to data preprocessing in Assignment 1, I give a detailed supplementary description and a description of feature selection here.

2.1 Data introduction

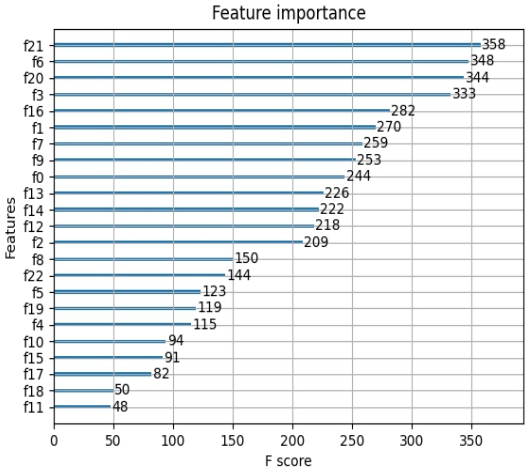
The training data we used were pre-processed by Xuanying Zhu et al. They ask 12 students with no prior knowledge of depression recognition were recruited to watch the full set of 16 German language depression videos. And 85 time domain features for each video watching session were extracted by them from those three physiological signals, Galvanic Skin Response, Skin Temperature and Pupillary Dilation. These features mainly focuse on capturing the amplitude variance and the occurrences of transient changes in the signals. In addition, Of the 85 features, 23 are from Galvanic Skin Response (GSR), 23 are from Skin Temperature (st) and 39 are from Pupillary Dilation (pd).

2.2 Data Prepare

I use the read_excel function in the pandas library to read three xlsx files: pupil_features.xlsx, gsr_features.xlsx, skintemp_features.xlsx. Then I used xgboost to carry out a simple classification model modeling, and selected the most influential features as the input to the neural network model, and discarded some of the lower influential features. The advantage of this is redundancy. The feature input of will make the model have the possibility of overfitting and reduce the generalization performance of the model, so feature selection is very necessary.

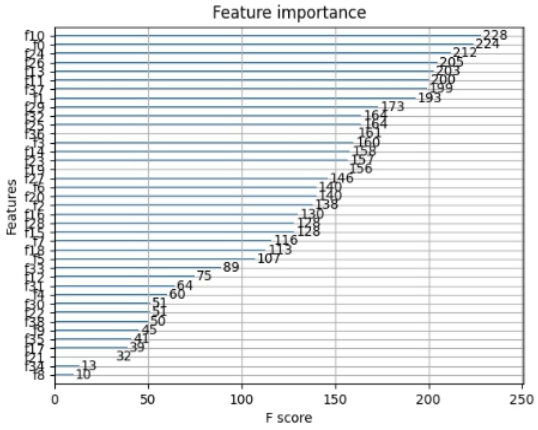
The following are my feature selection results for the three files pupil_features.xlsx, gsr_features.xlsx, skintemp_features.xlsx.And I discarded all the features with an importance score (F score) below 100.

2.2.1 gsr features



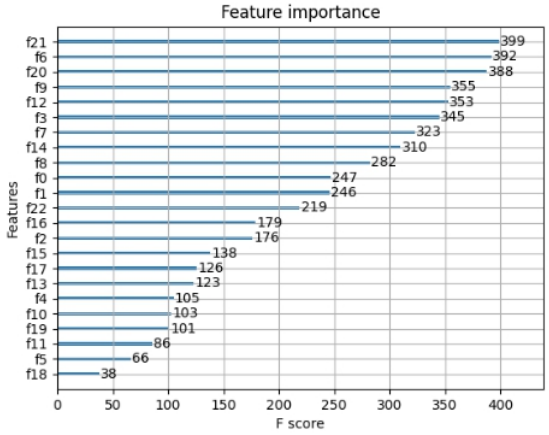
For the gsr feature, we only use the first 18 features, so the 11th, 18th, 17th, 15th, and 10th column features are discarded.

2.2.2 pupil features



For the pupil feature, we only use the first 26 features, so the 33h, 12th, 31th, 4th, 30th, 22th,38th, 9th, 35th, 17th,21th, 34th and 8th column features are discarded.

2.2.3 skintemp features



For the skintemp feature, we only use the first 20 features, so the 11h, 5th and 18th column features are discarded.

3 Method

3.1 Xgboost

Xgboost algorithm is generally used for modeling, but in my work, I use xgboost to establish a simple classification model for feature selection.

Xgboost is an improved algorithm based on GBDT algorithm. It combines the idea of the tree model and boosting algorithm, and is a kind of classification model (regression model) with excellent performance in machine learning.

The objective function of xbboost is composed of training loss and regularization term. The objective function is defined as follows:

$$Obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \qquad (1)$$

Where $l(y_i, \hat{y}_i)$ represents the loss function, $\Omega(f_k)$represents the regular term, the loss function is determined by the specific task, and the regular term represents the complexity of the tree, the essence of which is to restrict the tree from being too complex and avoid over fitting the model.

Let the loss function $l(y_i, \hat{y}_i)$Taylor expansion:

$$l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) = l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \quad (2)$$

By introducing the above second-order expansion into the objective function, the approximate value of the objective function can be obtained, and the constant term in the objective function can be removed：

$$\begin{aligned} Obj^{(t)} &\simeq \sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \\ g_i &= \partial_{\hat{y}^{(t-1)}} l\left(y_i, \hat{y}^{(t-1)}\right), h_i = \partial^2_{\hat{y}^{(t-1)}} l\left(\mathrm{y}_i \cdot \hat{y}^{(t-1)}\right) \end{aligned} \qquad (3)$$

The complexity of tree can be defined as:

$$\Omega(f_t) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \qquad (4)$$

Where T is the number of leaves, γ ， λ Represents the super parameter, w_j is the weight of the leaf.

The complexity of the tree is brought into the objective function to get the final objective function

$$\begin{aligned} Obj^{(t)} &= \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] + \gamma T \\ &= \sum_{j=1}^{T} \left[G_j w_j + \frac{1}{2} (H_j + \lambda) w_j^2 \right] + \gamma T \end{aligned} \qquad (5)$$

From the above formula, we can see that the objective function is a function of w_j.Therefore, it can be solved by the formula of maximum value of quadratic function of one variable

$$w_j^* = -\frac{G_j}{H_j+\lambda}, Obj = -\frac{1}{2}\sum_{j=1}^{T}\frac{G_j^2}{H_j+\lambda} + \gamma T \qquad (6)$$

Above w_j^* is the maximum value of the objective function w_j is the final solution.

3.2 Neural Network

All the neural networks we used are based on fully connected neural networks with a sigmoid hidden layer and an output layer of four output neurons, representing the four depression levels. The neural network can be expressed as eq7.

$$y = W_{out}\sigma(W_{hidden}x) \qquad (7)$$

The y is the prediction of the input x. The input x is a column vector whose rows are as many as the number of physiological signal' features we used. The σ means the relu activation function**(In assigment1 I used the sigmoid activation function, which is actually wrong, because sigmoid is generally used for two classifications) .**The W means the linear layer.

3.3 Training Strategy

We use cross-validation as my training strategy. But randomly splitting the data set will destroy the time continuity of the observer's physiological data, which will cause us to lose information about the physiological changes of the person in a continuous period. So we divide the data sequence into 12 equal-sized sub-samples in order, and use one sub-sample as the verification data, and the remaining 11 as the training data. In other words, the physiological data from one observer is used as the testing set, and the others' physiological data is used as the training set. The final result will be the average.

All the neural networks are trained with the Adam optimizer [4] using backpropagation with the Cross-Entropy loss function. **The learning rate of the Adam optimizer is set to 1e-3, beta1 is set to 0.9, and beta2 is set to 0.99.**

3.4 Fusion mode

We consider fusing the three physiological signals together to predict the degree of depression. We have adopted two methods in total.

The difference with assigment1 is that when we concat before input, we use a deeper and wider neural network for training. This is because concat before input, the size of the input will increase.

3.4.1 Concatenate At Input

Since We think the feature extraction and fitting capabilities of the neural network are strong enough, We conjecture that the neural network can automatically select useful features from those three physiological signals and filter out the bad features. So the input x can be expressed as eq8.

$$x = concat(x_{gsr}, x_{st}, x_{pd}) \qquad (8)$$

The x_{gsr}, x_{st} and x_{pd} represent the studen's three physiological signals in each iteration. And $concat(\cdot)$ means concatenate the given sequence of seq tensors by column.

3.4.2 Voting At Output

We think that all three types of physiological signal can actually classify individual's depression, so we can train three neural networks separately and then use the three types of physiological signal to predict for the same individual. Then we merge the three prediction results using an adaptive weighted voting method, which can be expressed as eq9.

$$y = z_{gsr} + \alpha z_{st} + \beta z_{pd} \qquad (9)$$

The α and β are the "nn.Parameter" which means can they can be learned automatically by the neural networks.

3.5 Evaluation Measures

To evaluate the quality of our model, we used precision, recall and F1-score as evaluation measures. Precision is based on our prediction results which is the percentage of the proportion of how many samples whose predictions are positive are truly positive samples. Its calculation formula can be expressed as eq10.

$$P = TP/(TP + FP) \qquad (10)$$

P means the precision, TP means the number of truly positive samples predicted by our model, FP means the number of samples which predicted to be a positive sample by model but in fact it is a negative sample.

recall is for our original sample, which is defined as how many positive examples in the dataset are predicted correctly. Its calculation formula can be expressed as eq11.

$$R = TP/(TP + FN) \qquad (11)$$

F means the recall, FN means the number of samples which predicted to be a negative sample by model but in fact it is a positive sample.

F1-score is a metric that comprehensively considers precision and recall, which takes the harmonic mean of precision and recall defined as eq12.

$$F1 = 2 \times (\text{Predicion} \times \text{Recall})/(\text{Precision} + \text{Recall}) \qquad (12)$$

It's worth emphasizing that all the evaluation measure is for a specific depression level. We calculated the average precision, recall and F1-score for all depression levels and different fuse method to give a view on the general prediction performance.

4 Result And Discussion

On the basis of assigment1, I used two fusion methods to experiment. The experimental results are shown in Table 1. "A" means "input connection" and "C" means "output vote"("B" means "middle concat",We did not conduct this experiment this time). This experiment is performed after feature selection, and for the A method, that is, input connection, the depth and width of the model are increased. We use a input_dims-32-16-8-4 network MLP structure for classification. For the model in the fusion method C, we also changed it to the structure of input_dims-16-8-4.

Table 1. The performance of update result

depression level	Fusing method	precision	recall	F1-score
None	A	0.87/0.51	0.79/0.49	0.83/0.50
	B	-	-	-
	C	0.91/0.85	0.87/0.83	0.89/0.84
Mild	A	0.83/0.48	0.79/0.45	0.81/0.46
	B	-	-	-
	C	0.87/0.84	0.83/0.81	0.85/0.82
Moderate	A	0.85/0.48	0.83/0.46	0.84/0.47
	B	-	-	-
	C	0.90/0.84	0.87/0.82	0.88/0.83
Severe	A	0.85/0.50	0.84/0.49	0.84/0.50
	B	-	-	-
	C	0.86/0.84	0.84/0.83	0.85/0.83

In the above table, the left side of "/" represents the results of this experiment, and the result on the right of "/" represents the experimental results of assist1. From the experimental results, we can see that the precision, recall, and F1-score indicators of this experiment are all higher The result of this time has been improved. This improvement is brought about by the feature selection performed by xgboost. After removing those redundant features, the model has better generalization performance and robustness for the problem of depression symptom classification. Secondly, for the input fusion method, the result this time is much better last time. This is because we use a deeper and wider neural network for the input fusion method, which improves the performance of the model. The shallower neural network has too few model parameters, making the final model under-fitting. Finally, the effect of voting at the output is still the best, but the advantage is not so obvious.

5 Conclusion and future work

From the results of my experiments, it can be seen that effective feature engineering will significantly increase the effect of model modeling. The feature selection method I used in this article is xgboost, but in fact, you can also manually design feature engineering by yourself, but this requires We have a deep understanding of the meaning of the feature itself and the background of the problem.

In future work, I hope to have the opportunity to try models other than neural networks, and combine these models through the idea of boosting. The result of multi-model boosting will make the final model have better generalization performance and robustness.

Reference

[1] Zhu, X., Gedeon, T., Caldwell, S., & Jones, R. (2019). Detecting emotional reactions to videos of depression. In INES'19: IEEE 23rd International Conference on Intelligent Engineering Systems (6 pp).

[2] Lian, Z., Li, Y., Tao, J., & Huang, J. (2018). Investigation of Multimodal Features, Classifiers and Fusion Methods for Emotion Recognition. ArXiv, abs/1809.06225.

[3] Chen T , Tong H , Benesty M . xgboost: Extreme Gradient Boosting[J]. 2016.

[4] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv Prepr. arXiv1412.6980, 2014.

[5] L.K.Miline, T.D.Gedeon, A,K,Skidmore, School of Computer Science and Engineering School of Geography The University of New South Wales