Neural Networks with the Threshold Adjustment Technique in Facial Emotions Classification

Zhuoxuan Jiang

Research School of Computer Science, Australian National University

Abstract. Neural Networks is a popular technique in classification problems. In our research, we develop a feedforward neural networks with threshold adjustment technique [2] for the binary classification. The novel idea of threshold adjustment is proposed by Milne, Gedeon and Skidmore(1995), it can simply adjust the balance of false positive and false negative classifications which are produced by neural networks [2]. To be more practical, we choose *Static Facial Expressions in the Wild (SFEW)* which include seven types of facial emotions as our data source which is very close to the real-world conditions [1]. We also develop the neural networks with the multi nodes output layer which is a more commonly used technique in classification problems as the baseline to evaluate the performance of our techniques. As the result shown in our experiment, the threshold adjustment on the single node output layer can slightly improve the performance in the binary classification problem of facial emotions when associating with convolutional neural networks (CNNs) and can be fit for different binary classification problems by using different criteria.

Keywords: Binary classification, Threshold, Convolutional neural networks (CNNs)

1 Introduction

The emotion classification is a practical topic for classification research. The database source *Static Facial Expressions in the Wild (SFEW)* we use is extracted from a temporal facial expressions database *Acted Facial Expressions in the Wild (AFEW)* which is extracted from movies, and *SFEW* provide an approximating real-world condition [1]. The research of *SFEW* uses the local phase quantization (*LPQ*) and the pyramid of histogram of oriented gradients (*PHOG*) descriptors and generates the first five principal components for each image [1]. In PHOG, each image is decomposed into sequence of cells at different levels of pyramid, and cells at different levels have different grid resolutions. Then PHOG descriptor is built by combining histogram of edge orientations of each cell to achieve more local discriminative representation of features than global measure [13]. PHOG was proposed by Bosch et al. [14] and has been efficiently used in object classification [13]. The LPQ descriptor is calculated based on short-term Fourier transform (SIFT) and uniform window function, and it is the method to use local phase to construct blur invariant features. The following figure shows the key steps of LPQ descriptor [12].

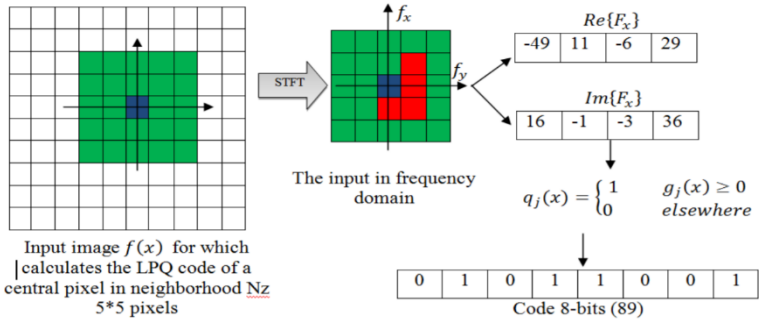

Fig. 1. Key steps of the local phase quantization (*LPQ*) descriptor.

The model trained in the paper uses a naive method (*nonlinear SVM* and the result on that dataset is not good(with 19% baseline classification accuracy) [1], so we choose this as our dataset because it still has much room to make progress and can bring a significant improvement on how good is the optimization. Additionally, it already generates the principal components and labels each image which can bring a lot of convenience on our research. Since the threshold adjustment technique is based on binary classification [2], we made a slight change on the labels of each image. We set the *disgust* as label 1 and all other emotions as 0, then the classification is transformed into binary classification. Then we conduct a classification problem of judging the emotion on the input image is *disgust* or others.

For the threshold adjustment technique [2], it introduces a method of breaking the balance of 0.5 as the threshold for classifying two classes on the single output unit. But a more popular way of classification is using multi units in the output layer, the number of units equals the number of classes, and each output unit represents the probability to be the

corresponding class. Then we compare the results of these two techniques under *SPI protocol* [2] and use *Precision, Recall* and *Specificity* as evaluation scores, additionally, we also use the result of the loss function and tables to analysis the performance.

For the comparison with the result from the research paper on *SFEW* and the multi nodes output layers technique, we use five principal features to represent each image as input to the naïve three-layer neural networks and average the accuracy on both two descriptors to be the performance of the model on this dataset. After the comparison and analysis on the dataset, we find new methods to make a better performance on the classification and better representation of the facial emotions, then we use the detected faces from raw images for convolutional neural networks(CNNs) and analysis how significant it can bring by adjusting threshold for binary classification. As the experiment result shown, the naïve three-layer networks perform not good due to the complexity of facial emotion images and the poor structure of the neural networks, but the adjustment of threshold can make an improvement when training with the well-structured networks and the effective face detection approach on the raw images.

2 Method

2.1 Preprocessing on the Dataset

Although images have already been extracted into five principal components, it still needs preprocessing to fill the null value. There is only one null value occurs, so we use the mean value of all images in that class to replace the null value. Another preprocessing method is rescaling input values. To reduce the chance of getting stuck in local minimum [5], we rescale input values with *StandardScaler* of *scikit-learn* which apply transformation with the mean and standard deviation of the dataset.

As we transfer the seven classes into binary labels, the dataset becomes imbalanced, and the prediction is prone to the class with more training patterns [6]. Thus, we use the approach of oversampling proposed by Buda, Maki and Mazurowski (2018) which add copies of existing patterns from categories with few training patterns [11]. After preprocessing, we split the dataset by two descriptors and train the model with the same parameters. Since our evaluation is under *SPI* protocol [1] that prediction is all on unseen objects, to fulfil the *SPI protocol*, we split the dataset into the training set, validation set and test set. We will use the training set to training models and predict on the validation set, then we can find the optimal parameters to use and use the optimal parameters to train the model and predict on the test set.

2.2 The Structure of the Neural Networks

Our first technique is a three-layer neural networks with two nodes in the output layer. This is a common and popular technique to use in classification problems. Each output represents the probability of the corresponding class, and the summation of all output node values is 1. Another technique is a three-layer neural networks with a single node output layer, then we classify the binary classes by setting the threshold. If the output value is bigger than the threshold, it predicts 1, otherwise predicts 0. As the result shown in the research [2], modifying the threshold can lead to a better performance in prediction. In another research by Kogan [3], the threshold can be treated as "100% separation and continues values for prediction value between 0 and 1, which seems to be a score.

When building the architecture of the networks, we first build a naïve three-layer networks based on backpropagation algorithm with one hidden layer and define five nodes input layer with five principal components of each image as input and a hidden layer with five hidden neurons. In the hidden layer, we choose *ReLu* as the activation function for both techniques. For the activation function in the output layer, we choose to use *Softmax* as the activation function and the *Cross Entropy* as the loss function in the multi nodes output layer, then the output node with the highest value(probability) is the prediction result. For the single node output layer, we use *Sigmoid* as activation function for the output unit and *BCE* as loss function which calculated the *Binary Cross Entropy* between the target and the output, however, to implement the threshold adjustment technique, the output value should be rescaling in zero to one, so we use *Sigmoid* as activation function to scale the output, if the value is bigger than the threshold, then the prediction is 1 which means the emotion is *disgust*, otherwise is 0 which means the emotion is others.

2.3 The Face Detection and CNNs

In our scenario, the original data source is the images captured in the movie and we use the *LPQ* and *PHOG* to extract the principal features as input, this process can reduce the input size and simplify the problem for neural networks. However, the descriptors we used are not robust enough for uncontrolled environment experiment [1] and this is proved to be a bottleneck to get a better performance, so we can also choose a better technique to describe and extract features from the image. Therefore, we implement the face detection using Haar feature-based cascade classifiers which is a popular object detection method proposed by Paul Viola and Michael Jones(2001) to extract the faces from each image, and using the detected faces to be the input of the networks. The approach has three key contributions, the first one is

representing the image with haar features, the second is an algorithm based on *Adaboost* to select features and yields classifiers, the third is combing complex classifiers with the cascade of classifiers [10].

Fig. 2. Sample detected faces using Haar feature-based cascade classifiers from the raw facial emotion images of *SFEW*.

Since the we use the detected images to be the input of the neural networks instead of the first five principal components, the naïve three-layer neural networks cannot deal with such complex situation. Thus, we developed the convolutional neural networks (CNNs) to be more well-structured. We use the LeNet-5 to be our structure of the convolutional neural networks (CNNs). The LeNet-5 is an eight-layer neural networks, which contains convolution, pooling and full connection layers and the structure can be represented as following figure [9].

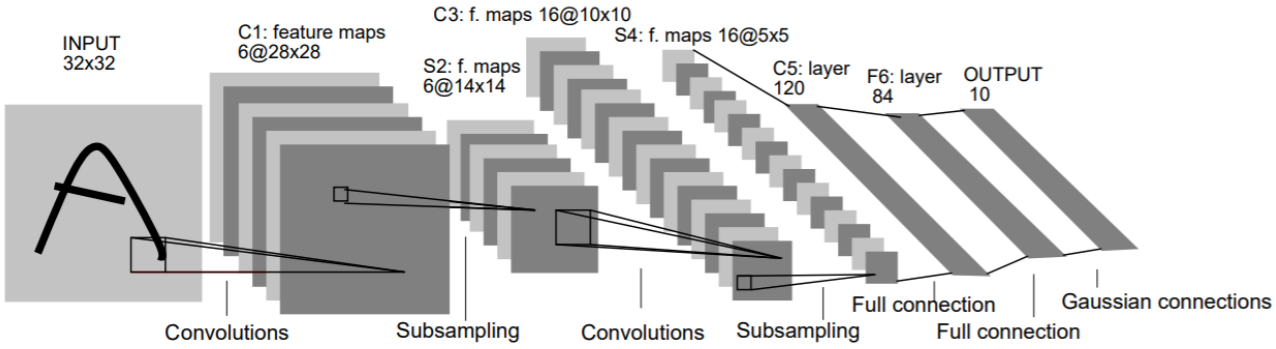

Fig. 2. Architecture of LeNet-5 from LeCun, Bottou, Bengio and Haffner (1998).

The pixels of each image are highly spatially correlated and using each pixel as individual input unit cannot deal with the correlation [9], but the convolution layer can help to extract spatial features according to the kernel size. Thus, CNNs is a meaningful deep learning technique to complete the classification task on facial emotion images. We also make slightly change on the structure of LeNet-5 according to the background of our classification problem. In the original structure of LeNet-5, it use 10 output units to detect ten classes of hand-writing numbers [9], however, in order to implementing the threshold adjustment techniques, we use single node output layer for binary classification.

The classification problem for the CNNs is still the binary classification of facial emotions, and the output of the networks is also associated with threshold adjustment technique, so we use the same activation function *Sigmoid* for output units, the same activation function *ReLu* for fully connected layers and the same loss function *BCE* as those of the previous three-layer neural networks.

2.4 Hyper-Parameters of the Neural Networks

For the optimizer, we choose to use *Adam* which is an extension of stochastic gradient descent (SGD) because it has less memory requirement and perform well in multi-layer neural network, and it also shows a lot of efficient performance in deep learning [4].

The key parameters for convolutional neural networks (CNNs) is the optimal epoch to stop training, while an early stop results in an insufficient training, and too many epochs to train may cause overfitting and result in a bad performance on the test set. since in the deep [15]. Additionally, the LeNet-5 would have very severe overfitting problems because of the huge size of networks [15]. Thus, we calculate the accuracy after each epoch and choose the optimal number of epochs for the convolutional neural networks (CNNs). The result is shown below, and we choose the 20 as the number of epochs.

Table 1. The loss on validation set when training with different number of epochs.

Epoch	1	3	5	10	15	20	25
Loss on validation set	1.532	0.638	0.458	0.085	0.047	0.27	0.40

Since the purpose of naïve three-layer neural networks is to compare the threshold adjustment technique and normal neural networks with the multi nodes output layer, thus, we use default hyperparameters for both techniques, then we choose to use 0.01 as the learning rate and 1000 as the optimal epoch for the three-layer neural networks.

2.5 The Threshold Adjustment Technique and SPI Protocol

The approach of adjusting threshold is proposed by Kogan (1991), in his research, he trained a classical backpropagation neural networks (BNN) to be an indicator to show whether the record is "GOOD" or "BAD", then he encode the two class as 0 and 1 and the value of the single output node is a continuous score between 0 and 1, and he propose a method to set a threshold as the border, when the output score is bigger than that value, it predicts them as "GOOD" otherwise predict them as "BAD", and it can help minimize the mean square error (*MSE*) by adjusting that threshold.

In our scenario, we implement this approach on judging whether a specific facial image belongs to "disgust" or not. To find the best threshold to implement, we make a list of different threshold values between 0 and 1, then loop all values to train the model and use the prediction performance on the validation set as metrics to find the optimal threshold. Since threshold is not part of the neural networks, we implement the process after training the neural networks. After training the model with the chosen optimal parameters and get the output value which is a continuous value between 0 and 1, then we use the threshold as a criterion to judge the output value belongs to which class.

As the research [2] shows that this approach can improve the performance of networks because modifying threshold can minimize the false-positive results. Thus, we calculate the *False Negative Ratio* as one of the metrics. *False Negative Ratio* should be calculated as follow:

$$\text{False Negative Ratio} = FP/ (FP + TN). \quad (1)$$

The *Accuracy* and *False Negative Ratio* of both descriptors and the convolutional neural networks (CNNs) are shown in following tables, then we use threshold 0.1 for both descriptors, and the optimal threshold on the convolutional neural networks (CNNs) is 0.95. However, the

Table 2. The value of threshold and corresponding *False Negative Ratio* and *Accuracy* on dataset from *LPQ*.

LPQ									
Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
False Negative Ratio	0.29	0.36	0.39	0.49	0.53	0.58	0.6	0.78	0.82
Accuracy	0.64	0.62	0.61	0.58	0.57	0.56	0.55	0.51	0.49

Table 3. The value of threshold and corresponding *False Negative Ratio* and *Accuracy* on dataset from *PHOG*.

PHOG									
Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
False Negative Ratio	0.19	0.58	0.58	0.61	0.68	0.7	0.72	0.77	0.79
Accuracy	0.67	0.58	0.58	0.57	0.54	0.53	0.53	0.51	0.50

Table 4. The value of threshold and corresponding *False Negative Ratio* and *Accuracy* of *CNNs*.

CNNs										
Threshold	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.95
False Negative Ratio	0	0	0.03	0.03	0.03	0.03	0.03	0.03	0.03	0.03
Accuracy	0.962	0.967	0.954	0.958	0.962	0.962	0.970	0.970	0.970	0.975

Since there are two descriptors in the dataset and we use them to train the model separately, we average the accuracy of models on both descriptors as the final score of each technique. To compare the performance of both techniques and the result from the *SFEW* research, we use *SPI protocol* [1] that prediction is all on unseen objects, and use the *Accuracy, Precision, Recall* and *Specificity* as evaluation scores. Additionally, we also use this technique to evaluate the performance of the convolutional neural networks (CNNs) with threshold adjustment. They are calculated as follow:

$$\text{Accuracy} = (TP + TN)/ (TP + FP + FN + TN). \quad (2)$$

$$\text{Precision} = TP/ (TP + FP). \quad (3)$$

$$\text{Recall} = TP/ (TP + FN). \quad (4)$$

$$\text{Specificity} = TN/ (TN + FP). \quad (5)$$

3 Result and Discussion

After implementing the above techniques and evaluation method, the performance of both techniques is shown in the following table. The accuracy is slightly higher with single nodes layer output layer. Although we choose the optimal threshold to minimize the *False Negative Ratio*, we can find the benefit that the adjustment of the threshold cannot bring significant improvement in the overall. The reason is that the threshold can only minimize the number of false negatives, if the *False Negative Ratio* is lower, as a tradeoff, the number of false positives will be higher, so there will be very small impact on the overall accuracy performance of the prediction. Moreover, since the condition of the dataset of *SFEW* is very complex [1], the structure of our neural networks is too simple for this complex environment, so the accuracy is still not perfect.

Table 5. Average *Precision, Recall* and *Specificity* results based on the *SPI protocol* of both techniques on the output layer and CNNs.

	Single Node Output Layer	**Multi Nodes Output Layer**	**CNNs with Threshold Adjustment**
Accuracy	0.637	0.63	0.99
Precision	0.645	0.62	0.98
Recall	0.62	0.70	1.0
Specificity	0.66	0.56	0.98

As the result shown in the research on the *SFEW* [1], it uses *non-linear SVM* and makes a seven classes classification, the evaluation method is the same as ours, the accuracy of the model is in table 4. We can see the accuracy is very low with *non-linear SVM* because of the complex nature of conditions in the database [1]. Since it doesn't show many details of *non-linear SVM* and has different classification background, we can't make any concrete comparison and conclude any advantage of our techniques, however, we can treat the result on the research as a baseline, thus, the result on the research shows that the complex condition in the dataset is also one of the reasons that our technique on the naïve three-layer neural networks is hard to get high accuracy.

Table 6. Average *Precision Recall* and *Specificity* results on the *SFEW* dataset based on the *SPI protocol* with *nonlinear SVM*.

Emotion	**Angry**	**Disgust**	**Fear**	**Happy**	**Neural**	**Sad**	**Surprise**
Precision	0.17	0.15	0.20	0.28	0.22	0.16	0.15
Recall	0.21	0.13	0.18	0.29	0.21	0.16	0.12
Specificity	0.48	0.66	0.64	0.51	0.61	0.60	0.66
Accuracy	0.19						

In order to analyze how significant the threshold adjustment can improve the networks, we make a more well-structured convolutional neural networks and reconstruct the input of networks to avoid the limitation of LPQ and PHOG descriptors, then the improvement with different threshold is shown as following plot, and we can see that the minimizing the *False Negative Ratio* cannot always brings improvement on the accuracy, as the result shown, the adjustment on threshold can perform better than normal default 0.5 threshold of binary classification, but the optimal criteria would be the total accuracy instead just focusing on the minimizing the *False Negative Ratio.* While the criteria can also base on the problem we are facing, if the false negative have much more weight than false positive then it can adjust the threshold to minimizing the *False Negative Ratio*, for example, if we judging whether the patient is sick or not, then false negatives are very dangerous, then we can minimizing the risk by adjusting the threshold.

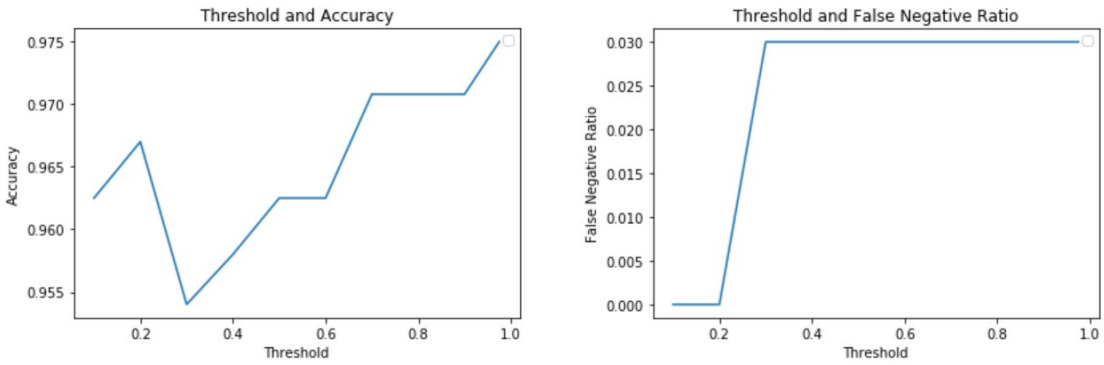

Fig. 3. Different thresholds and their performance on CNNs.

4 Conclusion and Future Work

We give two techniques in binary classification, which is the single node output layer and the multi node output layer in neural networks, and then make a comparison of their performance. Although the result of our experiment and implement does not bring a significant improvement than the baseline result from another research, but it is still a reasonable and valuable comparison which can be referred for researchers when facing binary classification. Especially, for the threshold in the single node output layer, it still has a lot of potential to be extended in many other situations. We only use the *False Negative Ratio* as criteria to choose optimal threshold, we can try more criteria to better evaluate the overall performance not only minimize the number of false positive.

To analyze how significant the threshold adjustment can improve the networks, we make more experiments on the structures of the neural networks to optimize the neural networks and try to get a better performance in the complex conditions. We use Haar feature-based cascade classifiers and LeNet-5 convolutional neural networks (CNNs), and we find the threshold adjustment technique can slightly improve the overall accuracy of classification, however, we also find that the threshold is a tradeoff between false positives and false negatives, and the criteria of choosing optimal threshold can be depends on the problem background.

However, for the threshold in the single node output layer, it still has a lot of potential to be extended. We only use the *False Negative Ratio* and the overall accuracy as criteria to choose optimal threshold, we can try more criteria to better deal with the different type of classification problems, and a statistical analysis on the distribution of threshold and its corresponding performance can help give a more reasonable and rigorous proof.

References

1. D.Abhinav, G.Roland, L.Simon, G.Tom: Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. Proceedings of the IEEE International Conference on Computer Vision. 2106-2112. 10.1109/ICCVW.2011.6130508, 2011.
2. M.LK, G.Tom, S.AK: Classifying Dry Sclerophyll Forest from Augmented Satellite Data: Comparing Neural Network, Decision Tree & Maximum Likelihood, 1995.
3. I.Kogan: Speculative Experiment With Neural Networks on Separation and Scoring in Financial Applications, IEEE Trans. on Neural Networks , 1991.
4. K.Diederik, B.Jimmy: Adam: A Method for Stochastic Optimization. International Conference on Learning Representations, 2014.
5. S.J., S.Joaquin: Importance of input data normalization for the application of neural networks to complex industrial problems. Nuclear Science, IEEE Transactions on. 44. 1464 - 1468. 10.1109/23.589532, 1997.
6. L.A., C.Alejandro, M.Alejandro, L.H.V.Ana: The impact of class imbalance in classification performance metrics based on the binary confusion matrix. Pattern Recognition. 91. 6829. 10.1016/j.patcog.2019.02.023, 2019.
7. S. Vani, T. V. M. Rao: An Experimental Approach towards the Performance Assessment of Various Optimizers on Convolutional Neural Network, 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, pp. 331-336, doi: 10.1109/ICOEI.2019.8862686, 2019.
8. L.Tuong, V.Minh, V.Bay, L.Mi, B.Sung: A Hybrid Approach Using Oversampling Technique and Cost-Sensitive Learning for Bankruptcy Prediction. Complexity. 2019. 1-12. 10.1155/2019/8460934, 2019.
9. Y. Lecun, L. Bottou, Y. Bengio and P. Haffner: Gradient-based learning applied to document recognition, Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998. Available: 10.1109/5.726791, 1998.
10. V.Paul, J.Michael: Rapid Object Detection using a Boosted Cascade of Simple Features. IEEE Conf Comput Vis Pattern Recognit. 1. I-511. 10.1109/CVPR.2001.990517, 2001.
11. M. Buda, A. Maki and M. Mazurowski: A systematic study of the class imbalance problem in convolutional neural networks, Neural Networks, vol. 106, pp. 249-259, 2018. Available: 10.1016/j.neunet.2018.07.011, 2018.

12. M.Belahcene, M.Laid, C.Ammar, O.Abdelmalik, B.Salah: Local descriptors and tensor local preserving projection in face recognition. 10.1109/EUVIP.2016.7764608, 2016.
13. S.Partha, M.B, D.Sachidanada: Pyramid Histogram of Oriented Gradients based Human Ear Identification Pyramid Histogram of Oriented Gradients based Human Ear Identification. International Journal of Control Theory and Applications. Volume 10. 125-133, 2017.
14. A. Bosch, A. Zisserman, and X. Munoz: Representing shape with a spatial pyramid kernel, Proceedings of the ACM International Conference on Image and Video Retrieval, 2007
15. A. Krizhevsky, I. Sutskever and G. Hinton: ImageNet classification with deep convolutional neural networks, Communications of the ACM, vol. 60, no. 6, pp. 84-90, 2017. Available: 10.1145/3065386, 2017.