Applying Pre-trained Convolutional Neural Network on Predicting SARS Symptoms: A Preliminary Research

Yuchen Fang

Research School of Computer Science,
The Australian National University,
Canberra Australia
U7100720@anu.edu.au

Abstract. Pre-trained convolutional neural network models have a widely application and achieved great success in image recognition and natural language process field. Most of them have a good performance in processing large scale of different datasets. This paper in order to increase the current SARS-CoV predicting task performance by utilizing a popular pre-trained convolutional neural network named AlexNet [4] and apply on the SARS-CoV-1 fuzzy signature dataset after fine-tuning the pre-trained model. Different from using multiple dimension data such as image and words, this paper shows a method of trying use one-dimension data as the input of model classifiers and get a well predicting result on the four labels classification problem. The accuracy is growing stably with the increase of training epochs and the average test accuracy is 98.45%, indicating great potential in applying multilayer convolutional neural network after fine-tuning in helping stop the current pandemic of COVID-19.

Keywords: Convolutional Neural networks, Fuzzy set, Fuzzy signature, SARS, Deep Learning, Fine-tune, COVID-19

1 Introduction

1.1 Background

SARS is a severe respiratory infectious disease, and have a rapid progress of respiratory failure in critical patients. It is highly infectious and with a rapid disease progress. To stop the epidemics of the highly infectious virus, increase capability in detecting the early cases is of great important. Features of SARS cases are important and useful to indicate SARS infection cases. It concludes: Body Temperature at different times of a day multiple parts, Systolic and Diastolic Blood Pressure, Nausea Symptom, and Abdominal Pain. The SARS-CoV-1 dataset includes these features in the form of fuzzy signature structure. The fuzzy signature structure [1] is a concept that can be used to find a relationship between complex structured data in classification tasks. There are many applications of fuzzy logic structure, one of it is in medical data processing. B. Mendis, T. Gedeon and L Kóczy (2005) [1] reported that the medical and economic diagnoses are the obvious applications of the fuzzy signatures. (p.1) Since most of the medical data are fuzzy and not easy to distinguish, the healthy index is a range, not a specific number. Fuzzy logic can use to describe the severe degree of various symptoms. Hospitals and doctors can use the fuzzy logic data to help diagnose the disease. In this way, the method of doing classification task based on fuzzy logic data is worthwhile, and it is the main target of this research.

1.2 Dataset

The SARS-CoV-1 dataset [1] contains four different kinds of cases and twenty-three column of fuzzy values. The four different cases are SARS, Hypertension, Pneumonia, and Normal cases. There are a thousand samples of each symptom. There is no label in the original dataset, and the name of file represent which class they belong to. Twenty-three columns of fuzzy values can be divided into four main parts, Temperature data, Blood Pressure data, Nausea symptom and Abdominal Pain symptom. The temperature data include four parts, recorded temperature of patients at 8am, 12pm, 4pm, 8pm four different time. The degree of temperature separated into three parts, Slight, Moderate and High degree at each record time, use the correlated fuzzy value to record. Blood Pressure is separated to two parts, Systolic blood pressure and Diastolic blood pressure, each of them has three degree, Slight, Medium and High. Nausea symptom is separated into Slight, Medium and High degree. Abdominal Pain symptom includes two kinds of fuzzy logic data, No or Yes. The brief structure of SARS-CoV-1 is listed in the Figure 1 as below.

$$A_S = \begin{bmatrix} \text{fever} \begin{bmatrix} 8\text{a. m} \\ 12\text{p. m} \\ 4\text{p. m} \\ 8\text{p. m} \end{bmatrix} \\ \text{blood pressure} \begin{bmatrix} \text{systolic} \\ \text{diastolic} \end{bmatrix} \\ \text{nausea} \\ \text{abdominal pain} \end{bmatrix}$$

Figure 1. The fuzzy structure of the SARS-CoV-1 dataset

2 Methods

2.1 Data Preprocessing

The features represented by fuzzy value in the dataset are range from zero to one. Applying normalization method to the fuzzy value can do the features scaling, make the data accords with the normal distribution. There are two popular data normalization method, Min-Max normalization and Zero-Mean normalization. Min-Max normalization have a widely use in the centralize sample data. Our fuzzy logic dataset is relatively dispersed, the Min-Max normalization is not feasible. Here we can use the zero-mean normalization method. It accords with the equation

$$x_n = \frac{x-\mu}{\sigma} \tag{1}$$

(μ is the mean of all samples,σ is the standard deviation of all samples). The four classes are labeled as the number zero to three, HighBP: 0, Normal:1, Pneumonia:2, SARS:3. Encoded classes with numbers is convenient for using convolutional neural network to finish consequent classification task. The means of twenty-three fuzzy logic data are zero and standard deviation are one. Eventually, the four separate datasets are assembled into one integrated dataset and the order of data is shuffled randomly. The ratio of training set, validation set and testing set is six to two to two. Because the pre-trained convolutional neural network has a large scale of parameters and hyperparameters, and we aim to find a suitable hyperparameter settings for our small dataset classification task. The validation set is useful in select the optimized the number of nodes in each layer and adjust the parameter to avoid overfitting and improve the generalization ability of the model. The test set is use to test the final performance of the model.

2.2 Pre-trained model architecture

The over all architecture of the pre-trained CNN named as AlexNet is depicted in Figure 2. [4] It is a complicated network contains eight layers with weights, and first five layers are convolutional and the remaining three are the fully-connected layers. The output of the last fully-connected layer is fed into a one-thousand-way softmax which produces a classifier do classification task over one thousand class labels. The AlexNet have two functional parts, features extraction part and the classifier part. The kernels of the third convolutional layer are connected to all kernel maps in the second layer. Krizhevsky, A., I. Sutskever , and G. Hinton(2012) The neurons in the fully-connected layers are connected to all neurons in the previous layer and the response-normalization layers follow the first and second convolutional layers. The Max-pooling layers follow both response-normalization layers as well as the fifth convolutional layer. The ReLU non-linearity is applied to the output of every convolutional and fully-connected layer.[4]

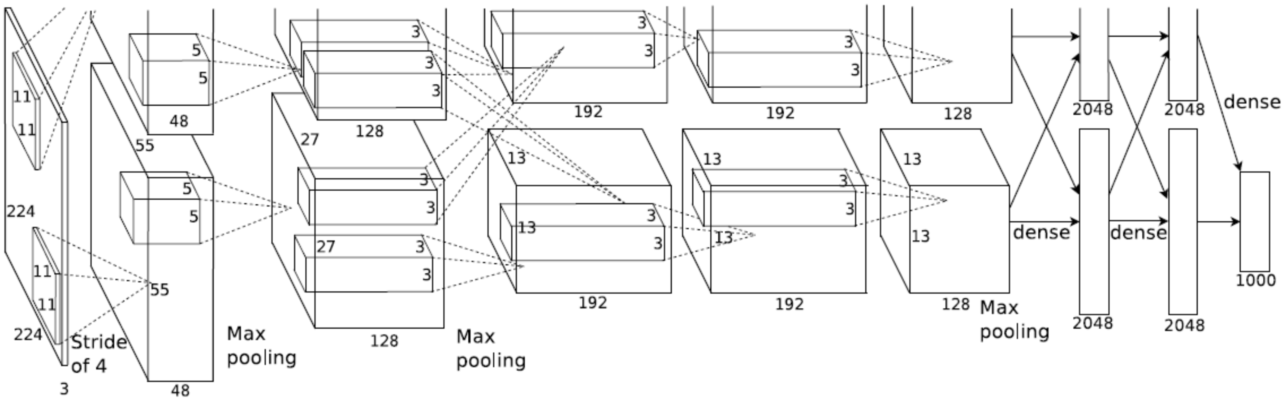

Figure 2. Brief illustration of the architecture of AlexNet

2.3 Fine-tuning on the model

Feature extraction is needed widely in image processing task, and we use one-dimensional fuzzy logic data as input. We can discard parts of layers on the feature extraction part, and reserve most of the layers on the classifier part. We can first do truncation on the last softmax layer, represent with a new non-linear layer. The original CNN have a one thousand classes layers, and we do classification task based on four classes, so we need to change the one thousand classes layers into four classes layers. Compared with initializing weights randomly, the weights of pre-trained model are relatively great, so we use a small learning rate at first to avoid modifying the weights in a great extent. We use a ten times smaller learning rate than the training from scratch learning rate. There are not many similarities between fuzzy logic data and multiple dimensional images. Freezing the parameters in the initial feature extraction layers and retrain the rest layers. It is of great important to retrain the last high layers according to the fuzzy input data.

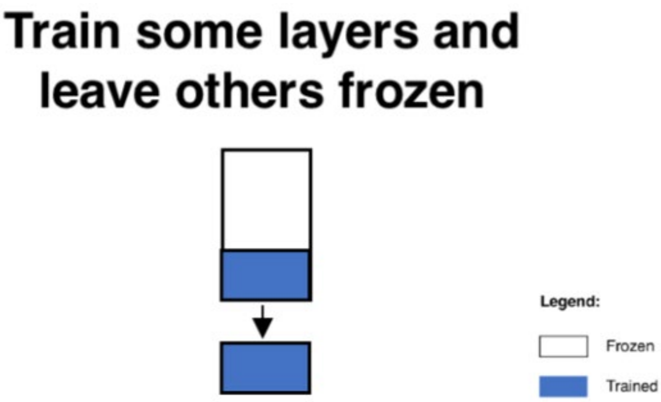

Figure 3. Model Freezing example

By freezing the multiple convolutional and MaxPool2d layers, we can add linear layer to match the input size of our twenty-three column fuzz logic data. By setting the probability of dropout layer as 0.4 after the ReLU activation layer, we avoid being overfitting on the training set and have higher average accuracy on the testing set.

3 Results and Discussion

After training and testing on the fine-tuning model, the overall performance is good, and have a great generalization ability on the independent testing set. In this part, we will analyze the results and show the feasibility of the model.

3.1 Effects of fine-tuning

Because we have mutually independent training and testing set, we have encountered an overfitting problem. The training accuracy is high (99% in average), but the performance on the testing set is bad (25.84% in average). Considering the task is a classification task on four labels, the 25% percent of accuracy is no better than a wild guess.

We can see from the confusion matrix, each case is distributed in a balanced way, which represent a bad generalization ability.

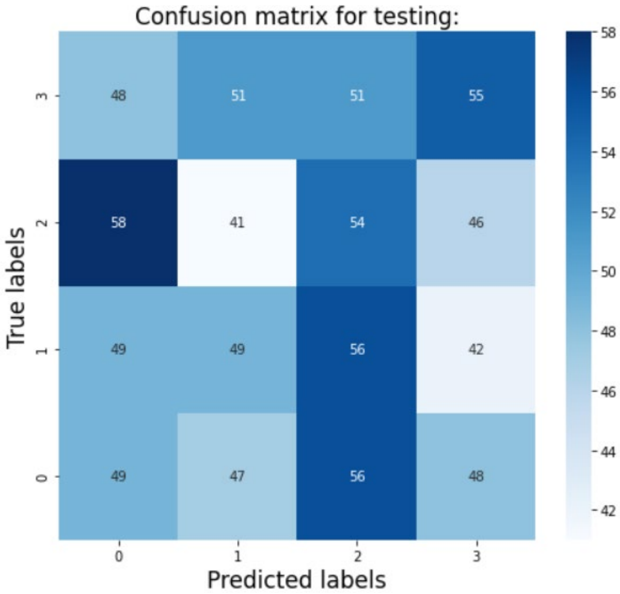

Figure 4. The overfitting confusion matrix

To solve this problem, I try to add a single dropout layer after the non-linearity ReLU layer, training and testing the model again. The training accuracy is decrease slightly (97% in average), but the performance on the testing set is increase noticeably (95.22%). This method can reduce the effect of overfitting in a great extent.

The confusion matrix shows the results, the dark blue color region is exactly on the diagonal of the matrix, with a few wrong predicted cases. The comparison of two confusion matrix shows the considerable effects of fine-tuning in adding layers to the pre-trained model.

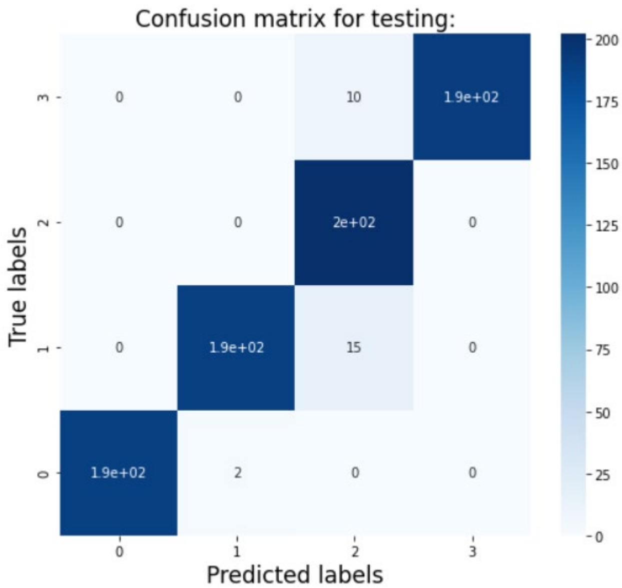

Figure 5. The Confusion matrix after adding layer

3.2 Performance of the model

The loss decreases continuously with the number of epochs increase, as the figure 5 depicts. The speed of loss convergence is slow at first and then grow faster in the latter epochs. I make multiple repetition training and each of the training have 550 epochs. It should be pointed out that, the losses are not same in each repetition and sometimes it reaches a value that is higher than other cases. It indicates that the convergence degree of loss is different, and it will affect the stability of the model.

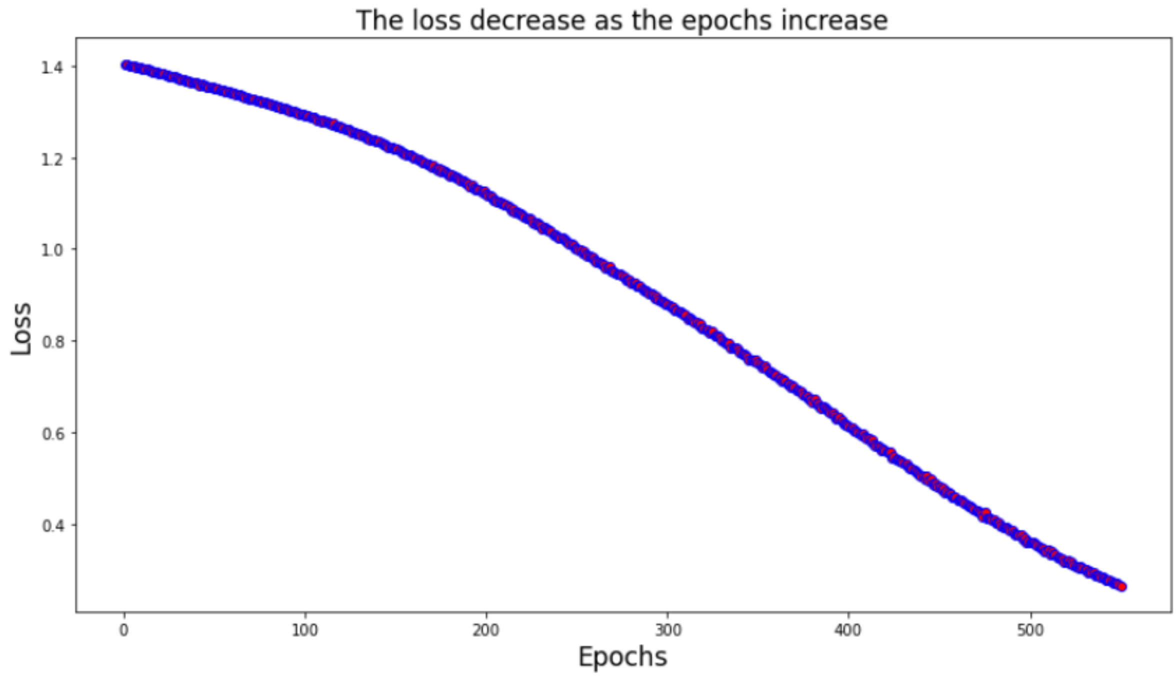

Figure 6. The loss decreases as the epochs increase

The accuracy increases significantly after 100 epochs and reach a stable peak after 300 epochs. It is potentially to modify the number of epochs to improve the training efficiency, because of the change of accuracy is nearly a straight line after 300 epochs.

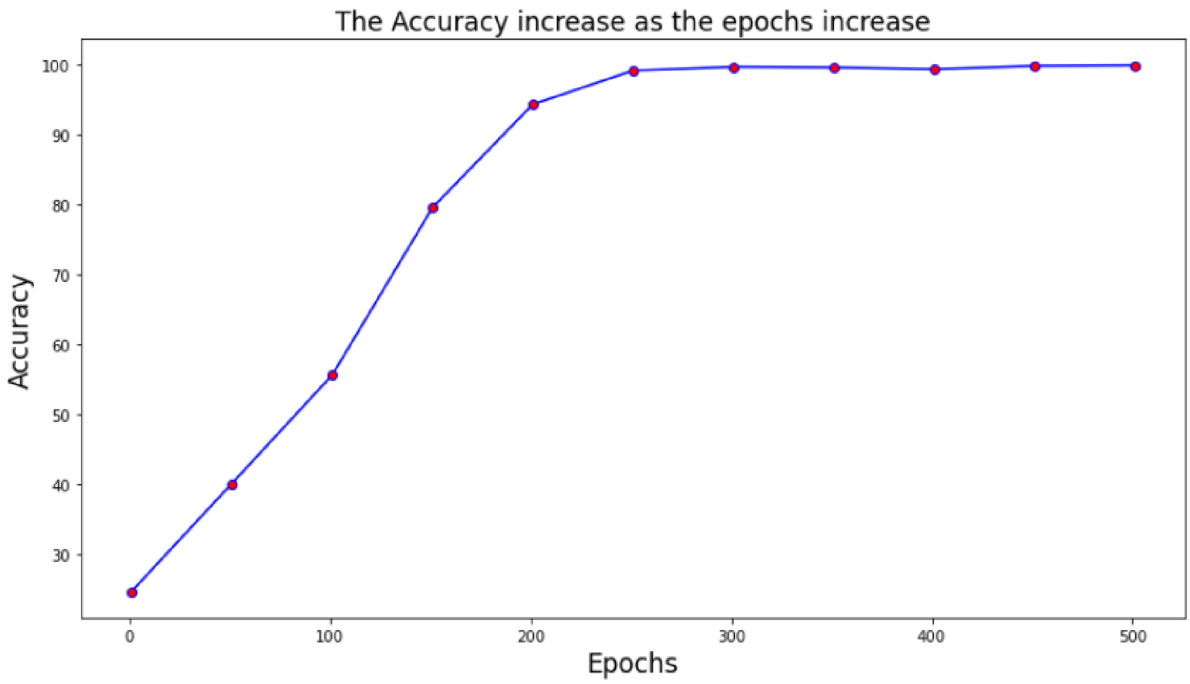

Figure 7. The accuracy increases as the epochs increase

The average testing accuracy is used to evaluate this fine-tuning CNN model. Running for 20 times and record the testing accuracy of each run, and sum them up to get the average testing accuracy. The average testing accuracy after 20 times is 95.22%, which is good in doing a four labels classification task.

3.3 Discussion

Since our target is finishing a classification task, we focus on the classifier part on the pre-trained model AlexNet, and there are different approaches can be followed to build a classifier. Comparing the performance of different classifiers on top of deep convolutional neural networks still requires further investigation and thus makes for an interesting research direction [16].

4 Limitations

There some limitations on the connection between pre-trained model and the fuzzy logic data. Pre-trained models are popular in applying on large scale datasets, but we have a relatively small dataset, and need to freeze lots of parameters to make the model fit for small dataset. The other problem is the principle of applying feature extraction operation on the fuzzy logic data is unknown, so it is difficult to decide which layer to freeze or retrain in the feature extraction part of AlexNet.

5 Conclusion and Future Work

In this paper, we applied a pre-trained model (based on AlexNet) on doing one-dimensional classification, and test on independent dataset. The method of applying CNN models on small dataset problem is feasible, it includes pre-trained model selection, freeze the redundant parameters on the convolutional layers, and add more layers such as dropout layer to fit the small dataset. We have 97% average training accuracy and 95.22% average testing accuracy which represent the generalization ability of model. We make a preliminary attempt in using pre-trained CNN models to do classification task in the SARS-CoV-1 dataset, and get some not bad results. There will be more accurate classification method on processing the fuzzy medical data in helping diagnose the disease. It also has great potential application in resisting the pandemic of COVID-19, which is the heartfelt wishes of people around the world.

References

[1] B. Mendis, T. Gedeon and L Kóczy, "Investigation of Aggregation in Fuzzy Signatures". 2005.

[2] Kok Wai Wong, T. Gedeon and L. Koczy, "Construction of fuzzy signature from data: an example of SARS pre-clinical diagnosis system," 2004 IEEE International Conference on Fuzzy Systems (IEEE Cat. No.04CH37542), 2004, pp. 1649-1654 vol.3, doi: 10.1109/FUZZY.2004.1375428.

[3] Canziani, A., Paszke, A. and Culurciello, E., 2016. An analysis of deep neural network models for practical applications. arXiv preprint arXiv:1605.07678.

[4] Krizhevsky, A., I. Sutskever , and G. Hinton . "ImageNet Classification with Deep Convolutional Neural Networks." Advances in neural information processing systems 25.2(2012).

[5] Weng, R., Yu, H., Huang, S., Cheng, S., & Luo, W. (2020). Acquiring Knowledge from Pre-Trained Model to Neural Machine Translation. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05), 9266-9273.

[6] Kandel, ER, Siegelbaum, SA and Schwartz, JH "Synaptic Transmission," Principles of Neural Sciences, 1991.

[7] V. Vukotić, C. Raymond, and G. Gravier. Bidirectional Joint Representation Learning with Symmetrical Deep Neural Networks for Multimodal and Crossmodal Applications. In Proceedings of the 2016 ACM International Conference on Multimedia Retrieval (ICMR), pages 343–346. ACM, New York, 2016.

[8] Gedeon, TD and Turner, H "Explaining studentgrades predicted by a neural network,"Proceedings InternationalJoint Conference on Neural Networks, pp. 609-612, 1993.

[9] Hecht-Nielsen, R "Counterpropagation Networks," Applied Optics, vol. 26, no. 3, pp. 4979-4984, 1987.

[10] Kosko, B. Neural Networks and Fuzzy Systems:A Dynamical Systems Approach to Machine Intelligence, Prince-Hall, Inc., Englewood Cliffs, NJ, USA, 1992.

[11] Quinlan, JR C4.5: Programs for Machine Learning, SanMateo, California: Morgan Kaufmann, 1993.

[12] Rumelhart, DE, Hinton, GE and Williams, RJ "Learning internal representations by error propagation," in Rumelhart, DE, McClelland,*"Parallel distributed processing,"* Vol. 1, MIT Press, 1986.

[13] Hora, N, Enbutsu, I and Baba, K "Fuzzy rule extraction from a multilayer neural net," Proc. IEEE, vol. 2, pp. 461-465, 1991.

[14] Kandel, ER, Siegelbaum, SA and Schwartz, JH “Synaptic Transmission,” Principles of Neural Sciences, 1991.

[15] Kosko, B “Bidirectional Associative Memories,”IEEE Transactions on Systems, Man, and Cybernetics, vol. SMC-18, pp. 49-60, 1988.

[16] Rawat, W. and Wang, Z., 2017. Deep convolutional neural networks for image classification: A comprehensive review. Neural computation, 29(9), pp.2352–2449