National Tsing Hua University
Fall 2023 11210IPT 553000
Deep Learning in Biomedical Optical Imaging
Homework 2

CHIH-YING,LIEN[1]

[1] *Institute of Photonics Technologies, National Tsing Hua University, Hsinchu 30013, Taiwan*

Student ID:110066520

1. Introduction

In this report, we will construct an Artificial Neural Network (ANN) for classifying chest X-ray images as either pneumonia or normal. Our dataset comprises both abnormal and normal images. To evaluate our training model's performance, we will employ both Binary Cross-Entropy Loss (BCE) and Cross-Entropy Loss (CE) for classification. Additionally, we will tune hyperparameter to train our model and enhance its performance.

2. Task A: *Performance between BCE loss and CE loss*

2.1 Introduction

BCE Loss and CE Loss are two commonly used loss functions in machine learning, the former is primarily used for binary classification tasks where each example belongs to one of two classes: positive (1) or negative (0). The latter is suitable for the multi-class classification tasks, where there are more than two classes. Therefore, we use BCE loss with sigmoid function to predict the probability of one of two classes, which the loss is minimized when the predicted probability aligns with the true binary label. As for the CE loss, we use softmax function because it transforms model outputs into class probabilities, ensuring the loss measures the difference between predicted and true class distributions.

2.2 Results and Analysis

To testify the performance between BCE loss and CE loss, we used same model architecture, which both dropout is 0.7 and the number of epochs is 10, other hyperparameters remain the same as the Lab 2. Fig. 1 (a) and (b) shows the model accuracy and model loss trained in BCE loss, respectively, while Fig 1 (c) and (d) shows the model accuracy and model loss trained in CE loss, respectively. The results indicate a test accuracy of 80.75% when using BCE loss and 73% when using CE loss. In my perspective that BCE loss model shows better test accuracy than CE loss is because BCE loss with sigmoid is designed for binary classification and models the probability of an example belonging to one of the two classes (e.g., 0 or 1). However, CE loss with argmax will choose the maximum value, potentially causing the model to be overly confident, especially when it encounters uncertain cases during training.

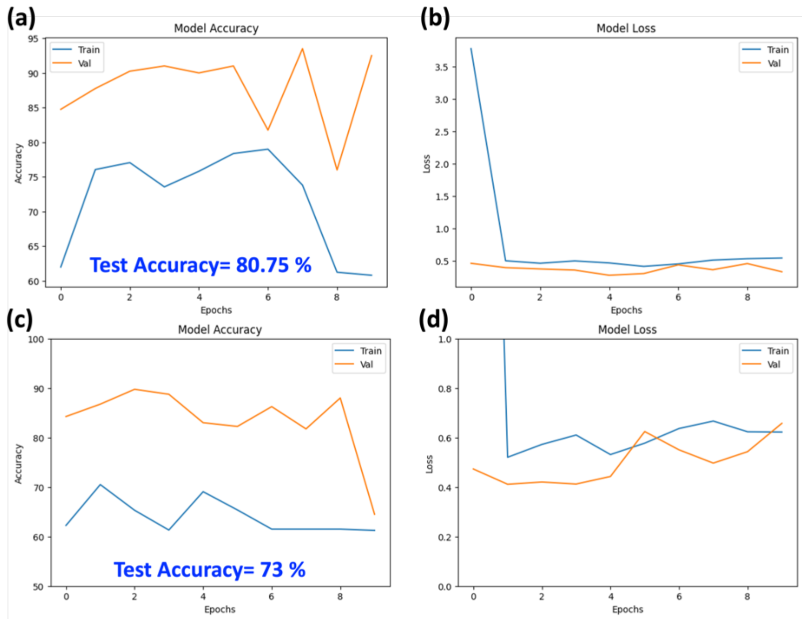

Fig. 1 (a) and (b) represent the model accuracy and model loss trained in BCE loss, respectively. (c) and (d) represent the model accuracy and model loss trained in CE loss, respectively

3. Task B: Performance between Different Hyperparameters

3.1 Introduction

The performance of deep learning models critically depends on hyperparameters, such as batch size, learning rate, layer depths, and dropout. Adjusting these hyperparameters to enhance model performance and reduce the risk of overfitting or underfitting, achieving more accurate and robust predictions. In this task, I have changed three hyperparameter: batch sizes, dropout and the number of epochs.

3.2 Results and Analysis

First, I experimented with varying batch sizes, using values of 2, 16, 32, and 64 as shown in Fig. 2 (a) to (d), respectively. The results showed that test accuracy consistently ranged between 72% and 76%, which don't show much optimized performance. However, despite these changes in batch size, I noticed that the test accuracy remained significantly lower than both the training and validation accuracy. In my point of view, the train model might be overfitting.

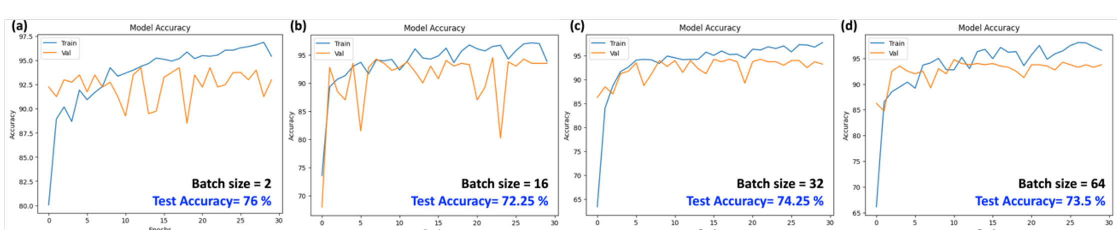

Fig. 2 The model accuracy between different batch size: (a) 2, (b) 16, (c) 32, and (d) 64.

To demonstrate that the train model may be overfitting, I decided to add dropout in my neural network model. Dropout is a regularization technique that reduces model complexity by randomly deactivating some neural network nodes during training. I tested dropout values of 0.3, 0.5, and 0.7 as shown in Fig. 3 (a) to (c), respectively. We can find that with higher dropout as 0.7, the test accuracy is optimized to 78.75%. Moreover, when adding the dropout, the train accuracy become lower than validation accuracy, this is quite normal because dropout is enabled during training, causing the model becomes sparser, meaning that only a subset of neurons is active for each training iteration. During validation, dropout should typically be disabled so that the model can use all neurons for predictions, resulting in more accurate results.

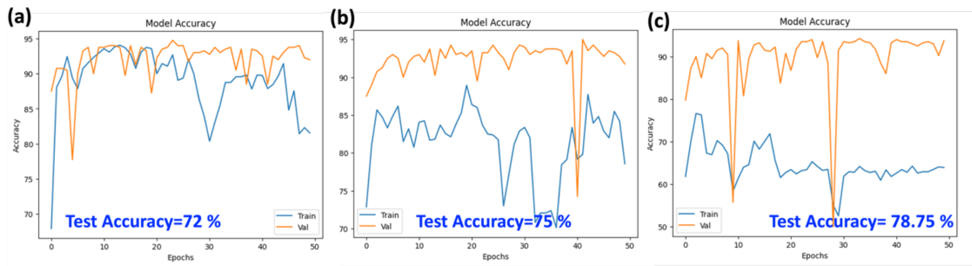

Fig. 3 The model accuracy between different number of dropout: (a) 0.3, (b) 0.5, and (c) 0.7.

In addition to adding dropout to avoid overfitting, there are other hyperparameters that can be optimized for training the model, such as the number of epochs. Consequently, in the next step, I conducted experiments with varying numbers of epochs: 5, 10, 25, and 50, as shown in Fig. 4 (a) to (d). The results revealed that the model achieved the best training performance with 10 epochs, reaching an optimized test accuracy of 82.5%. This outcome is attributed to the fact that when training a deep learning model, an increase in the number of training epochs such as 50 epochs could potentially lead to overfitting. On the contrary, If the number of training epochs is not enough such as 5, the train model could not well train causing underfitting.

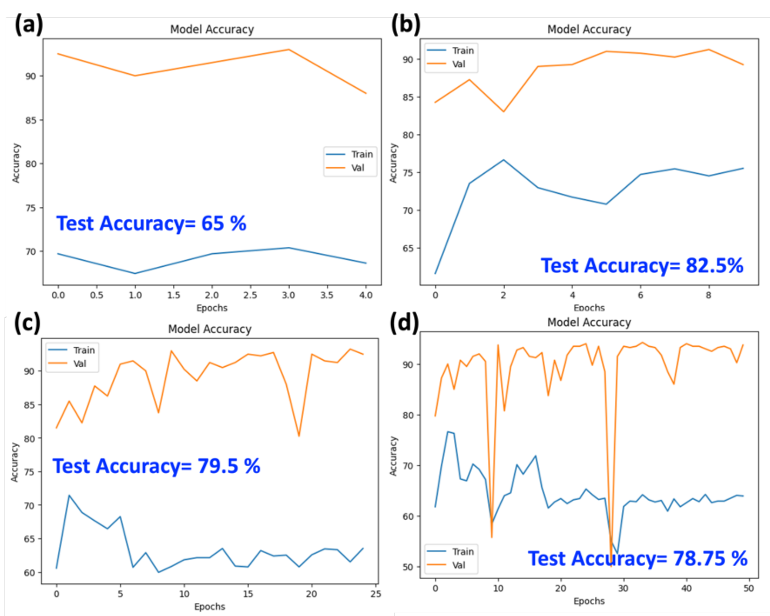

Fig. 4 The model accuracy between different number of epochs: (a) 5, (b) 10, (c) 25, and (d) 50.

Overall, the combination of a 0.7 dropout rate and 10 epochs yielded the most favorable results. To validate this finding, I conducted these two setting to test additional three times, and the results consistently showed that the test accuracy remained above 81%, shown in Fig. 5 (a), (c) and (e), and the Figures 5 (b), (d), and (f) demonstrate that both training and validation loss is around 0.5, indicating favorable results.

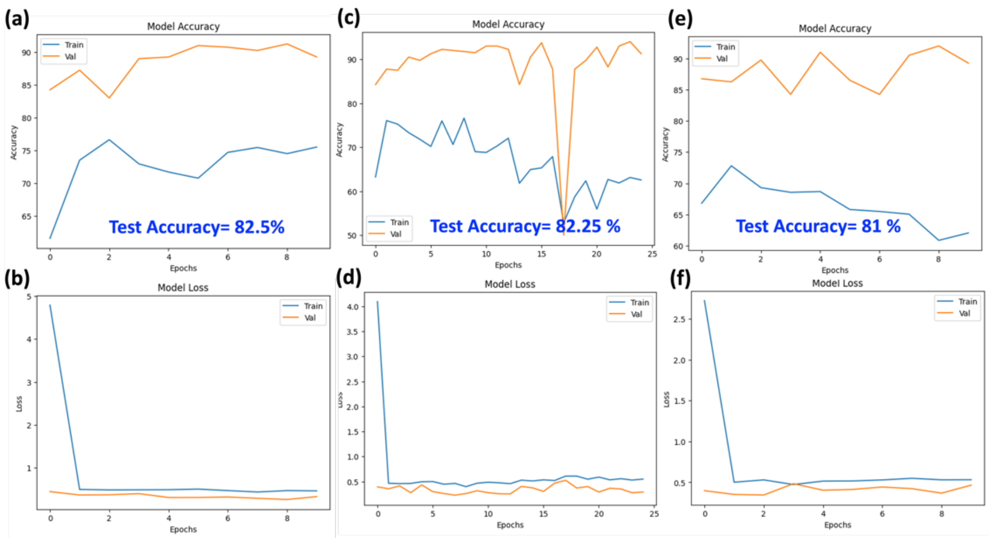

Fig. 5 (a), (c) and (e) is the model accuracy and (b), (d) and (e) is the model loss of different round.

4. Conclusion

In the task A, comparing BCE and CE loss with the same model configuration, both using dropout 0.7 and ten epochs, BCE performed CE with a test accuracy of 80.75% versus 73%. In my point of view, BCE) with sigmoid is ideal for binary classification as it models the probability of an example belonging to one of two classes, while CE with argmax can sometimes lead to overconfidence in classification.

In the task B, I think the test accuracy is lower than train accuracy is primarily attributed to model overfitting. To address this issue, I added dropout in the model and adjusted the number of training epochs. As a result, when a dropout rate of 0.7 was employed, the test accuracy improved to 78.75%, surpassing without dropout and with 0.3 and 0.5 dropout. Overall, the model achieved its best test accuracy of 82.5% with 10 epochs.

Overall, after doing this homework, I have realized more about how to build up a deep learning model, including choosing loss functions, optimizer and activate function , tuning the hyperparameters, and the meaning of each codes.