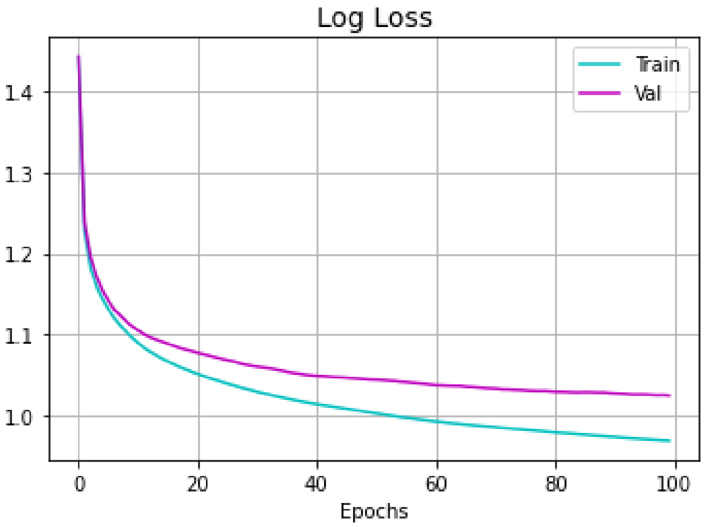
ML_HW#3

106033212 曾靖桐

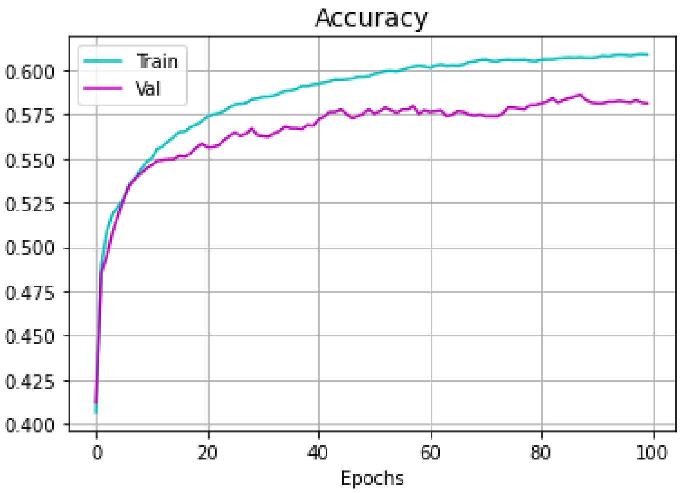
1.

(a)



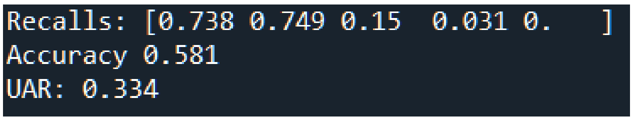
As epochs increases, training loss keeps decreasing, and so does the validation loss. However, the validation loss is higher than the training loss because the model may be specialized to training data.

(b)



As epochs increases, training accuracy keeps increasing, and so does the validation accuracy. However, the training accuracy is higher than the validation accuracy because the model may be specialized to training data. By observing the two plots above and comparing the value between these two sets, I think this model is a good fit because there is not a huge difference between validation accuracy (0.581) and training accuracy (0.609).

(c)



```
Recalls: [0.738 0.749 0.15  0.031 0.    ]
Accuracy 0.581
UAR: 0.334
```

Class-1: 7064

Class-2: 8195

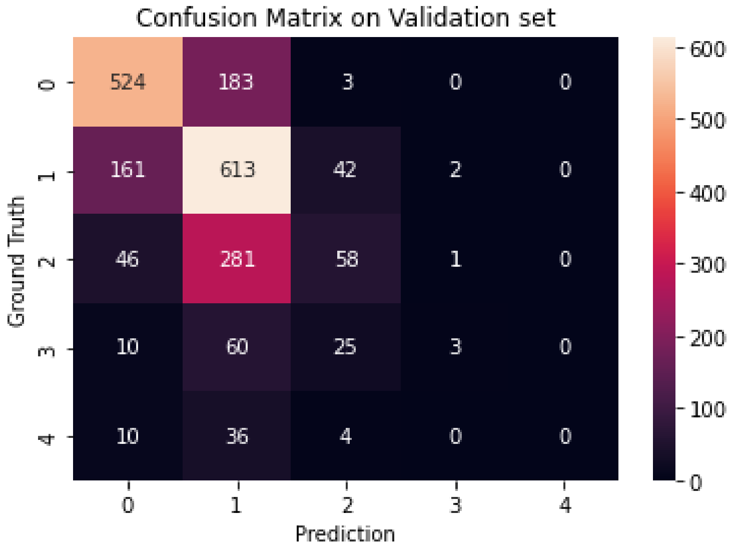
Class-3: 3855

Class-4: 1031

Class-5: 476

The accuracies of each class differ a lot because the target classes have an uneven distribution. Thus, the model would be more likely to predict data to the majority class.

(d)



Ground Truth \ Prediction	0	1	2	3	4
0	524	183	3	0	0
1	161	613	42	2	0
2	46	281	58	1	0
3	10	60	25	3	0
4	10	36	4	0	0

By observing the confusion matrix, most of the data are classified into class-1 and class-2 which have larger number of samples. The imbalanced data causes the model to be more likely to predict data as the first two classes.

2.

(a)

	(16, 16, 16)	(16, 16, 16, 16)	(16, 16, 16, 16, 16)
100	Time of training: 00:14 Loss: 1.025 Accuracy: 0.581 UAR: 0.334	Time of training: 00:16 Loss: 1.009 Accuracy: 0.598 UAR: 0.35	Time of training: 00:19 Loss: 1.035 Accuracy: 0.594 UAR: 0.347
200	Time of training: 00:26 Loss: 1.027 Accuracy: 0.594 UAR: 0.342	Time of training: 00:32 Loss: 1.012 Accuracy: 0.612 UAR: 0.37	Time of training: 00:36 Loss: 1.049 Accuracy: 0.591 UAR: 0.353
400	Time of training: 00:57 Loss: 1.029 Accuracy: 0.592 UAR: 0.345	Time of training: 01:03 Loss: 0.993 Accuracy: 0.613 UAR: 0.378	Time of training: 01:11 Loss: 1.101 Accuracy: 0.598 UAR: 0.362

-Best performing classifier (epochs:100, hidden_layer_size: (16, 16, 16, 16)):

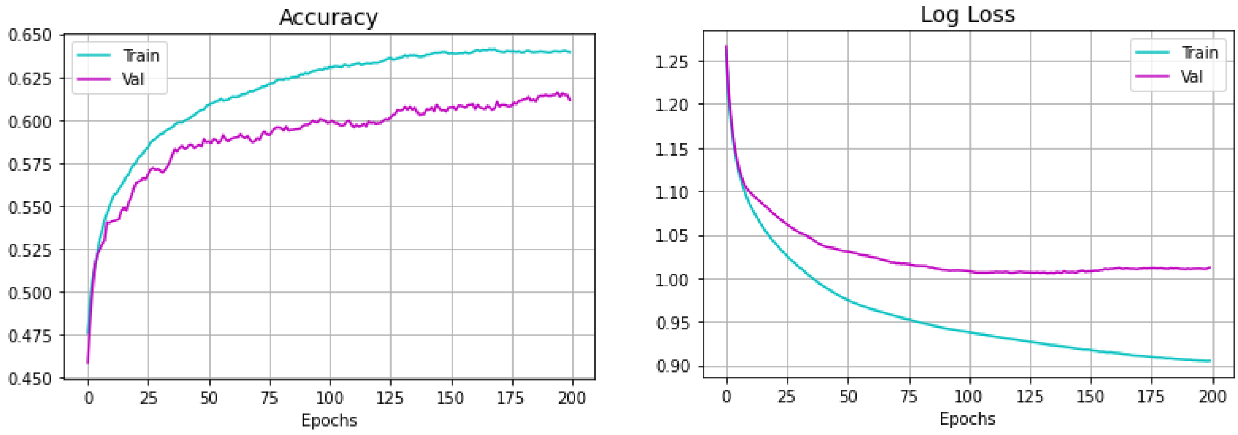
I think this is the best performing classifier because it has relatively high accuracy, high UAR, low loss, and short process time of training. Moreover, the model isn't overfitting.

-Worst performing classifier (epochs:400, hidden_layer_size: (16, 16, 16, 16, 16)):

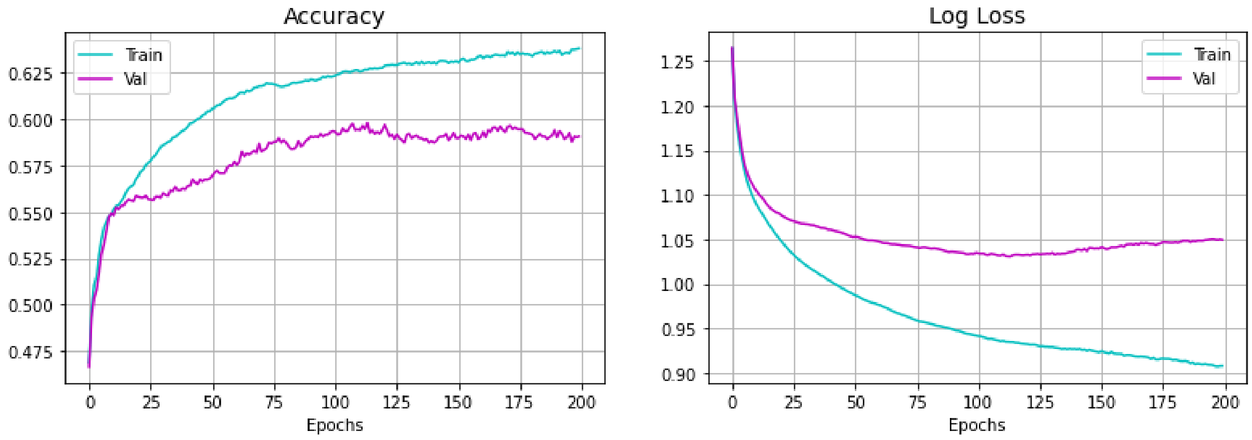
I think this is the worst performing classifier because it has the longest process time of training. Though it has relatively high UAR, it has the highest loss. Moreover, it's an overfitting model because of the deep neural network.

(b)

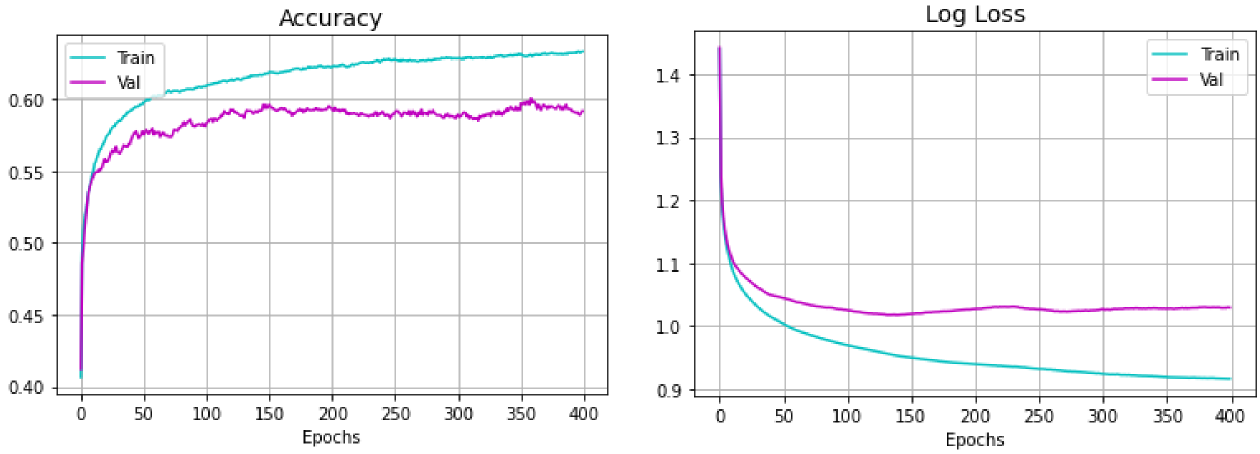
-Epochs: 200, hidden_layer_size: (16, 16, 16, 16)



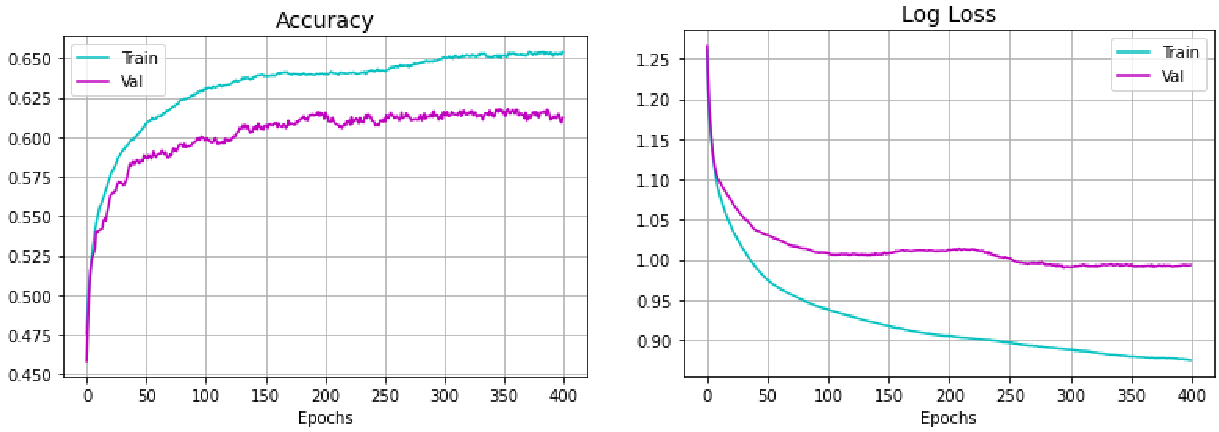
-Epochs: 200, hidden_layer_size: (16, 16, 16, 16, 16)



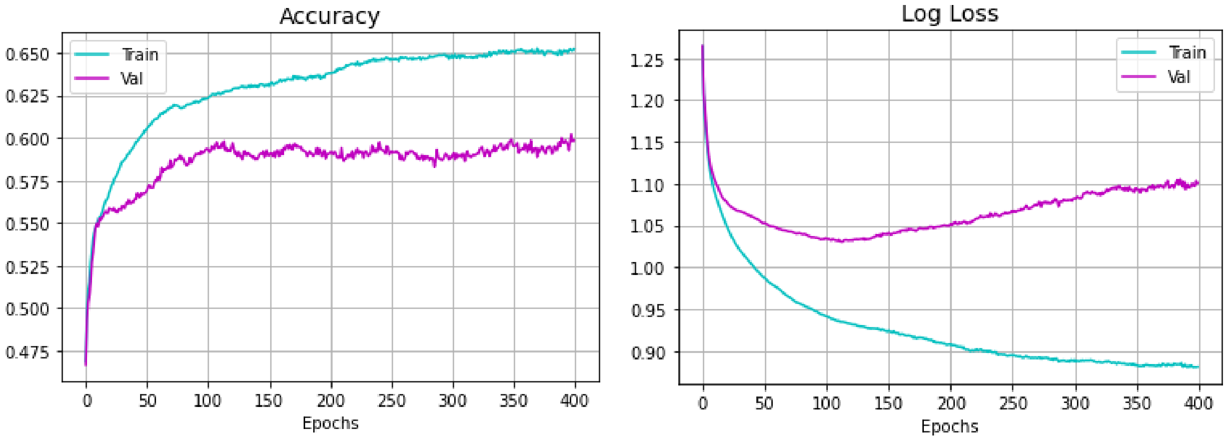
-Epochs: 400, hidden_layer_size: (16, 16, 16)



-Epochs: 400, hidden_layer_size: (16, 16, 16, 16)

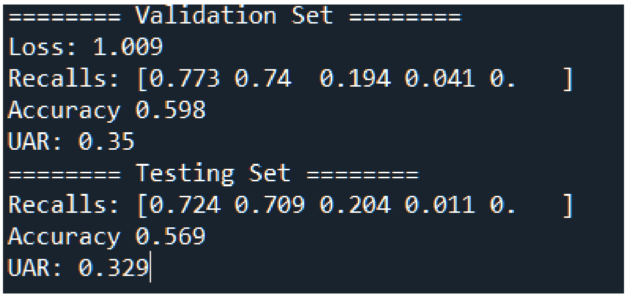


-Epochs: 400, hidden_layer_size: (16, 16, 16, 16, 16)



As the training loss keeps decreasing, the validation loss remains the same and starts to increase. This means that the model is too specialized in solving training data, so the validation loss would remain the same or get worse.

3.



```
======== Validation Set ========
Loss: 1.009
Recalls: [0.773 0.74  0.194 0.041 0.   ]
Accuracy 0.598
UAR: 0.35
======== Testing Set ========
Recalls: [0.724 0.709 0.204 0.011 0.   ]
Accuracy 0.569
UAR: 0.329
```

Yes, I think these scores as competitive as those evaluated in the validation set. The testing set scores are only 3-5% higher than validation set scores.

Using the validation set can check whether the model is overfitting or underfitting. That is to say, we can predict how the model might perform before testing. It can effectively prevent the condition that the model is highly specialized in training data, or the model doesn’t fit the data well. Both conditions would reduce the testing accuracy.