Thank you hsien rong, now I will share on the hyperparameter tuning analysis for learning rate, epochs and batch size

Starting with the learning rate, this hyperparameter is by far the most important one when it comes to building a robust model. Setting the learning rate too high will prevent the model from reaching close to the minima loss. This happens as it will oscillate left and right of the minimum which is also known as overshooting. This is observed in the moving average loss graph on the right where it is extremely noisy. There also can be times when the LR is too high that the model ends up diverging instead. As shown graph moving upwards in the classification task.

Setting the learning rate too low will prevent the model from converging, given the fixed number of epochs for our task. As you can see in the diagram on the left , the moving loss average has still yet to plateau. This means that the model has still yet to learn from the training loss. This is evident in our results as shown on the left with the training loss being significantly higher than the benchmark models. Also, for other tasks when the problem becomes non-linear, setting too low a learning rate will also cause the model to be trapped in a local minima. Essentially, finding the “sweet spot” for learning rate is vital .

Moving on to the epochs… Having too low an epoch will naturally lead to underfitting as shown the in diagram. This is because the model has yet to fully learn from the training loss.

having too high an epoch can be detrimental to the model as it could cause overfitting. Note that we did not really observe this much due to the epochs cap for our assignment. However, after a certain point, the rate of decrease in training loss become really small and this diminishing return might not be favourable.

Lastly, for the batch size, we found that setting too low a batch size value will cause accuracies to drop significantly. This is because smaller batch size has more noise and also more variance which all leads to unstable training. On the other hand, too high a batch size has little to no impact on the results and is faster per iteration. However, this is very straining on the hardware as it consumes a lot of memory.

Therefore, we felt that batch size should be a hyperparameter that is tuned last and only catered based on processing power of the device used.

Now I will handover to jay.