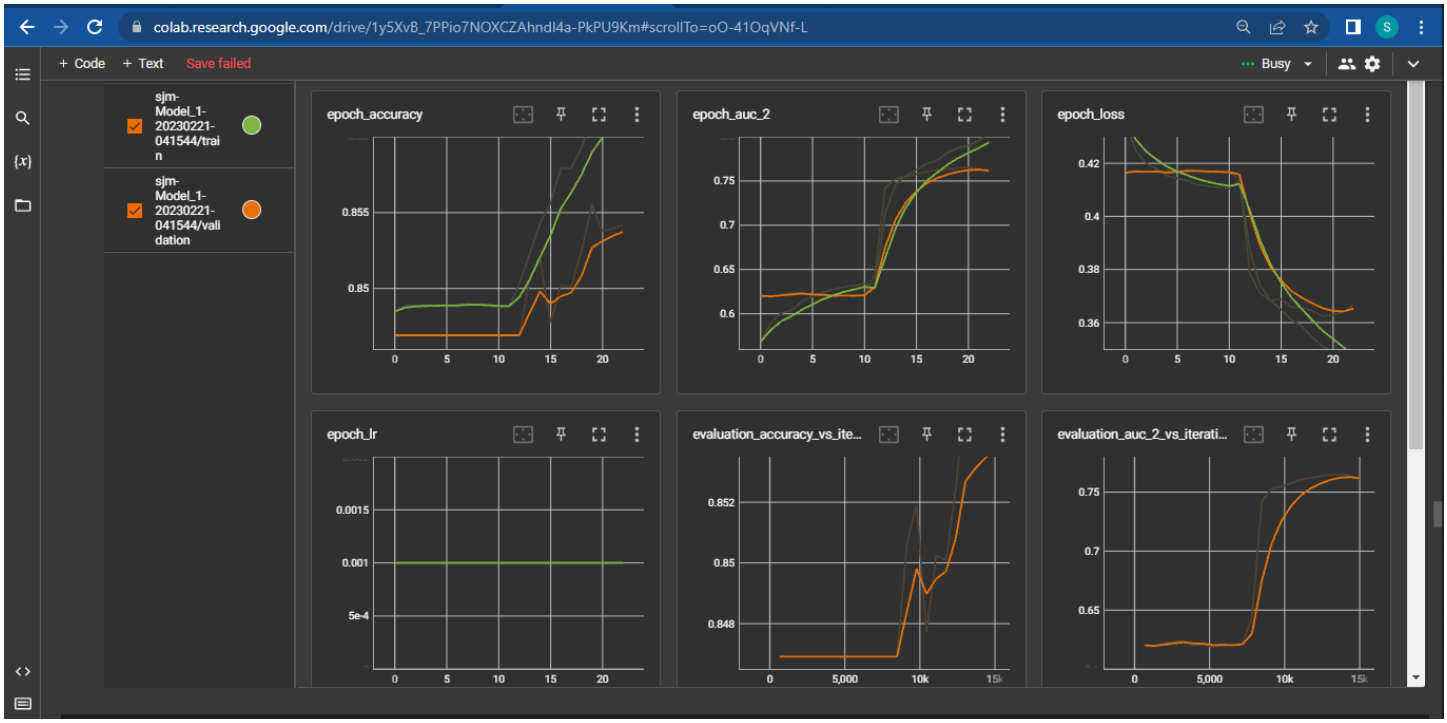
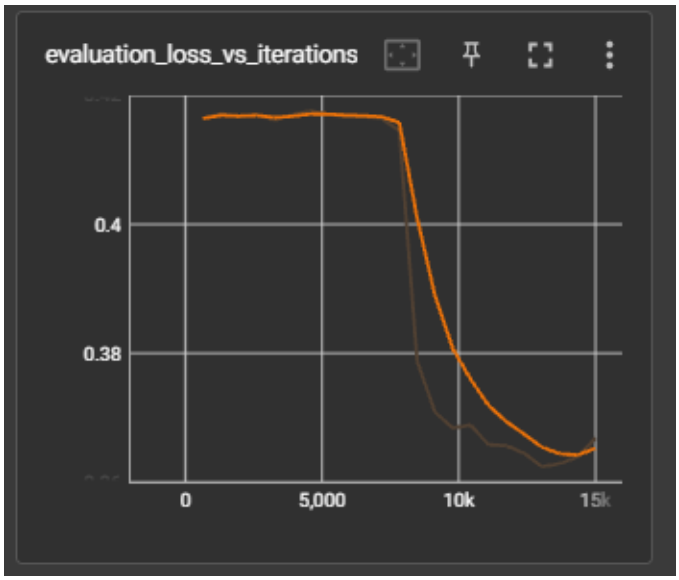
LSTM On Donors Choose

Model-1

Scalars:





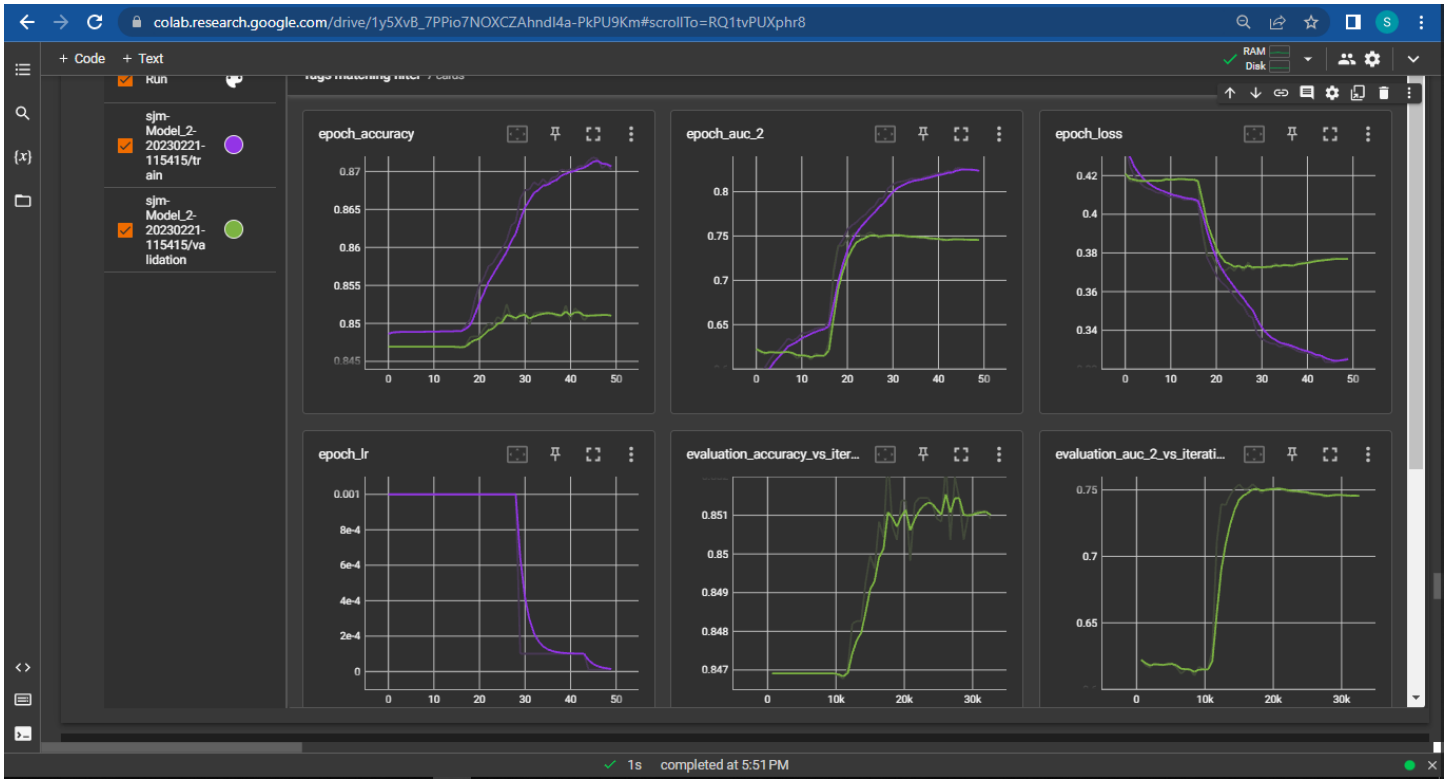
From the epoch accuracy and epoch_AUC plots we can see that both the train and test scores are constant to begin with but later increase suddenly after the ~10th epoch. This can be inferred as that the model was stuck in some kind of a saddle/local minima point and was able to get out of it after the 10th epoch.

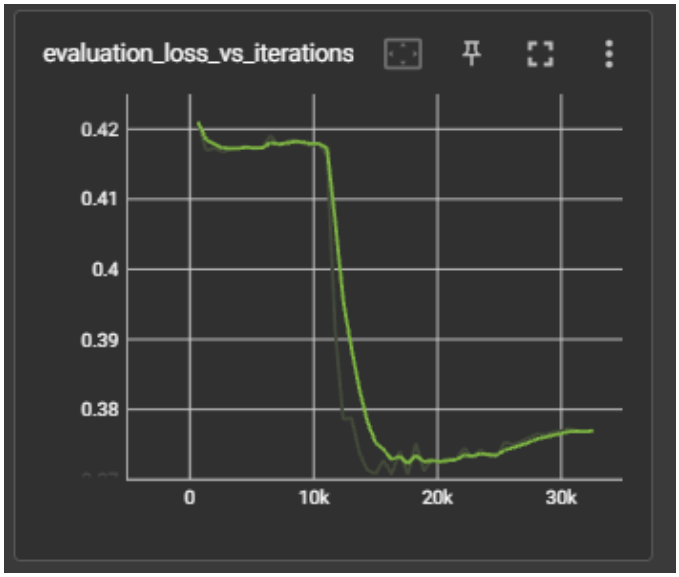
The same can be inferred from the epoch loss plot as well as the other plots.

Also if we take a closer look at around the 20th epoch the values start to plateau and also the gap b/w the train and test curves starts to increase. This can be seen as the start of the overfit stage.

Model 2

Scalars:



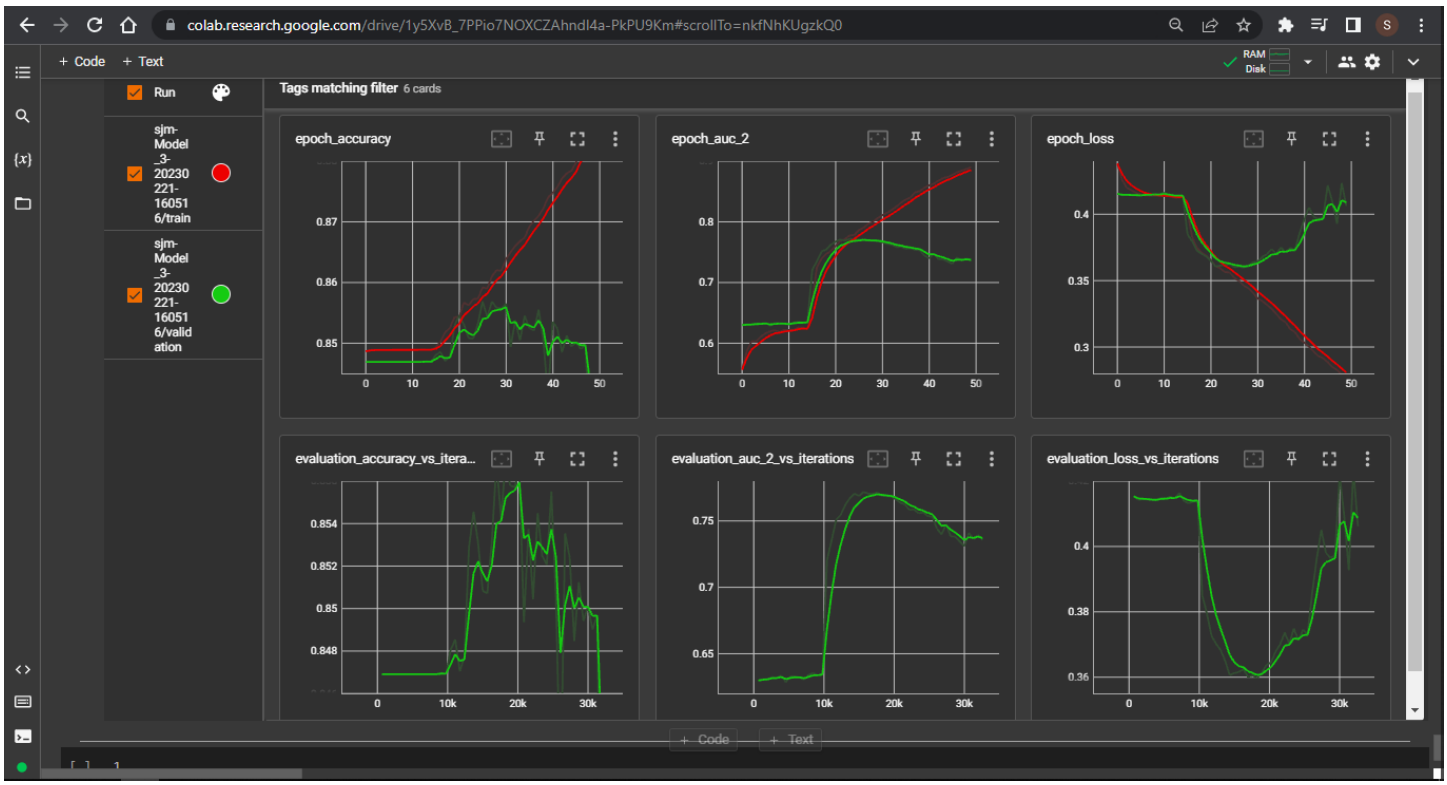


Same pattern as model-1 follows here.
From the **epoch accuracy** and **epoch_AUC** plots we can see that both the **train and test scores are constant** to **begin** with but **later increase suddenly** after the **~16th epoch**. The same pattern can be seen from the **epoch loss** plot as well.

Also if we take a closer look at around the **23rd epoch the values start to plateau** and also the **gap b/w the train and test curves starts to increase**. This can be seen as the **start of the overfit stage**

Model 3

Scalars:



Epoch Accuracy:
From the above plot we can see that the train and validation accuracy are constant till the 16th -17th epoch and after that train accuracy shoots upwards whereas the test accuracy first increases for some epochs and then starts to decrease.

Epoch AUC:
The train AUC first increases and plateaus around epoch 12 and then again shoots up after that. Similarly for the test AUC the same happens, first its constant and then increases and then again starts to decrease slowly after around the 25th epoch.

Epoch loss:
The train loss decreases first and then plateaus till the 15th epoch and after that keeps decreasing rapidly after that. The test loss at first is constant till the 15th epoch and then decreases till the 25th epoch and starts to increase after that.

From all the above plots we can infer that the model initially learns some things from the data, but then gets stuck at some saddle/local minima point due to which the plateau shows up in the earlier stages of the plots. But then it gets out from that to learn further and the weights start to update, hence the performance of the model increases.

But looking at the model after the training stage we can say that it's kind of in an overfit stage. Training up-to / around 25 epochs would have been optimal.