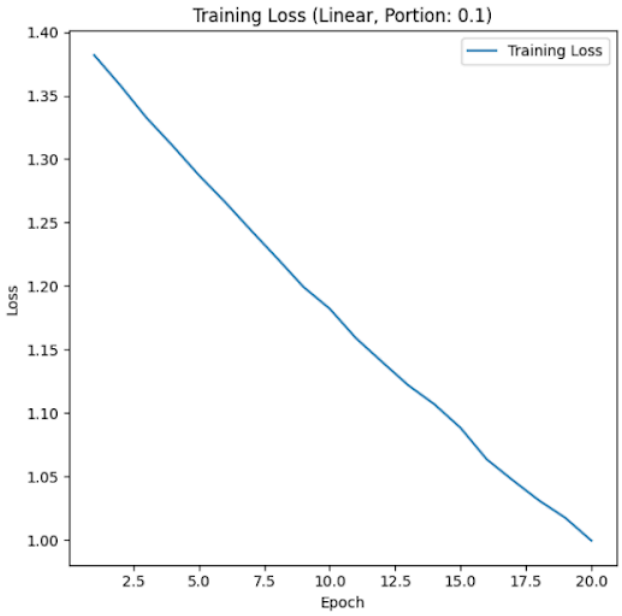
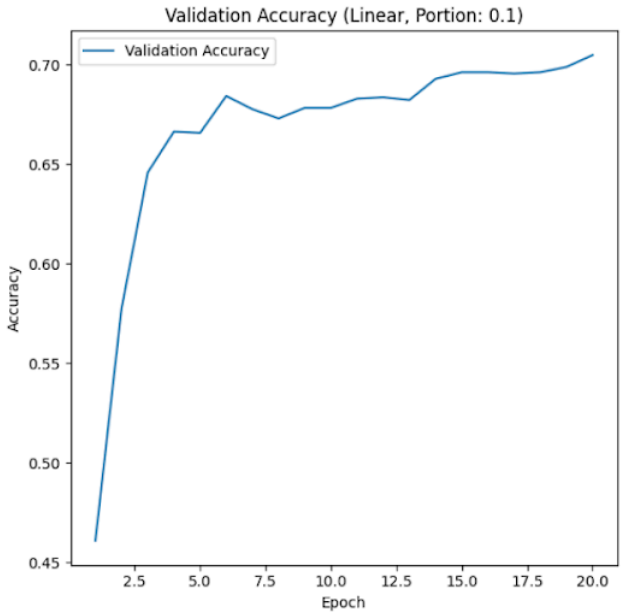
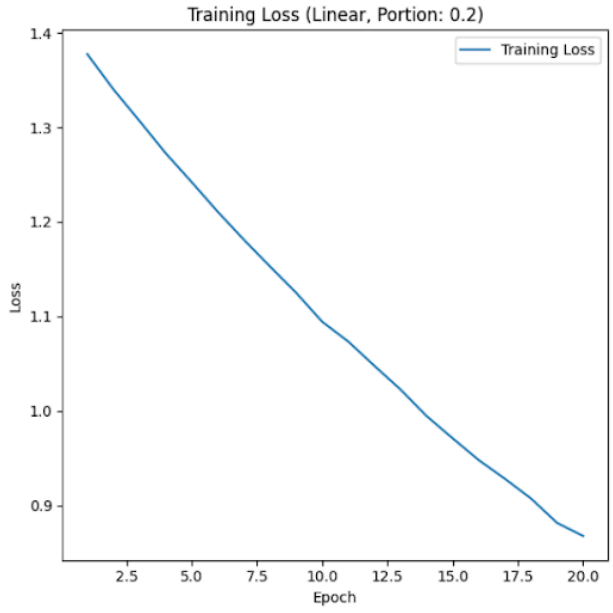
Natural Language Processing– Ex2

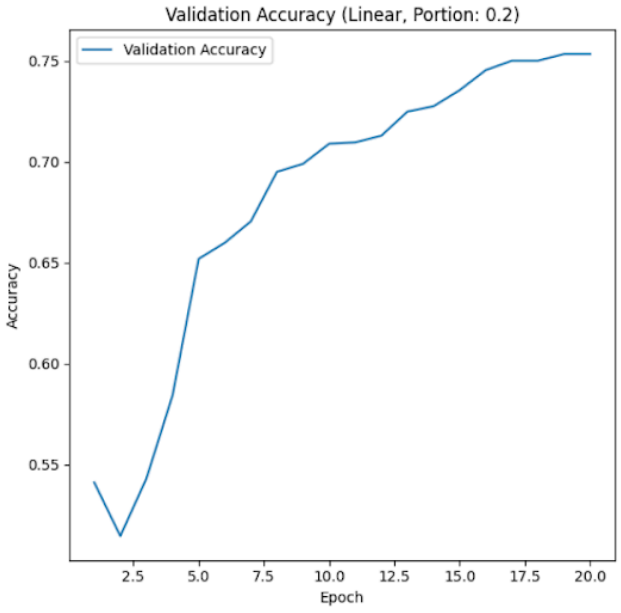
Zenab Waked, Malak Laham

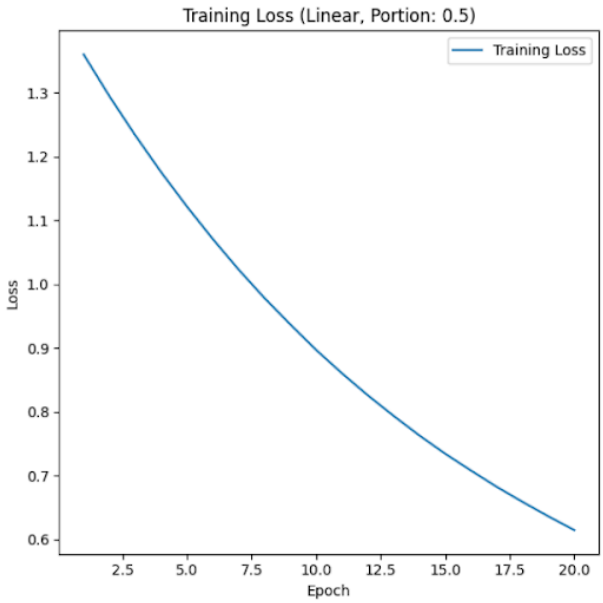
Task 1 :

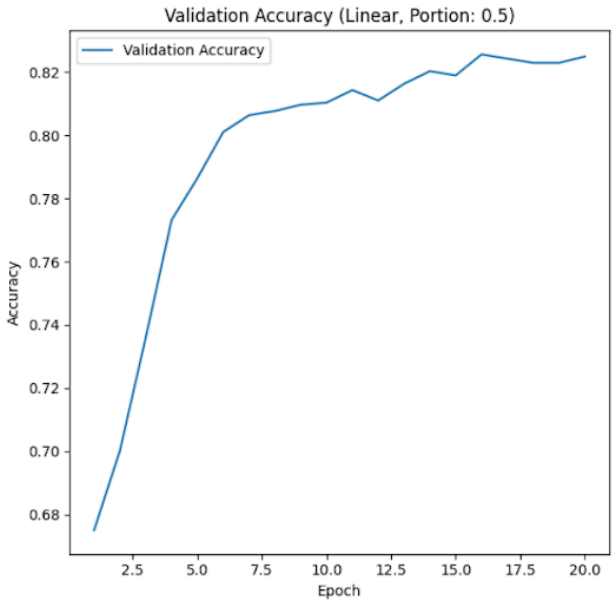


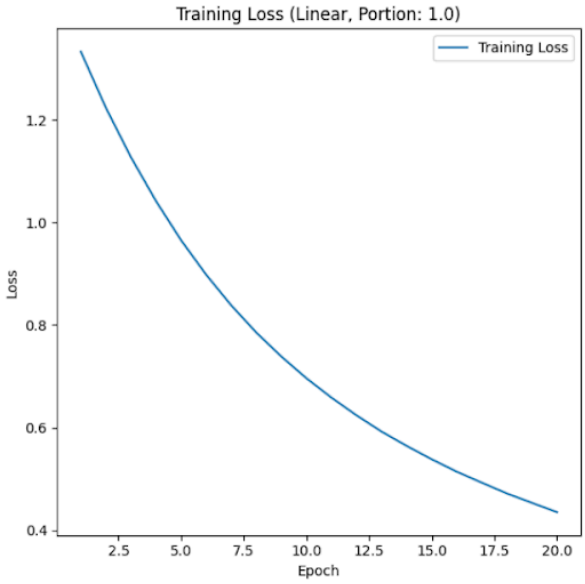


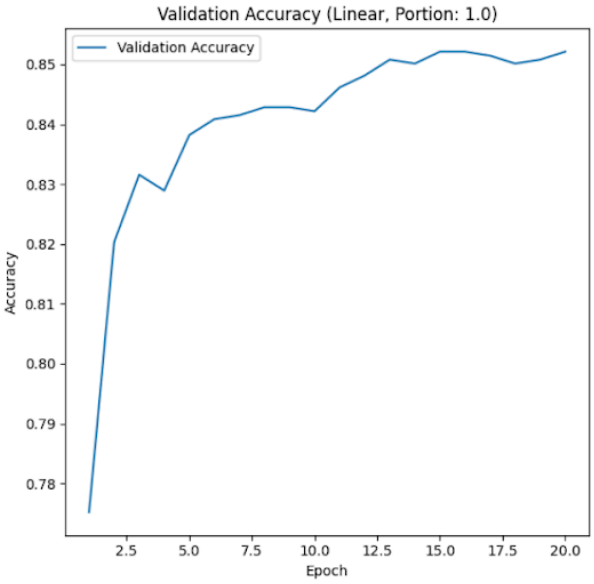




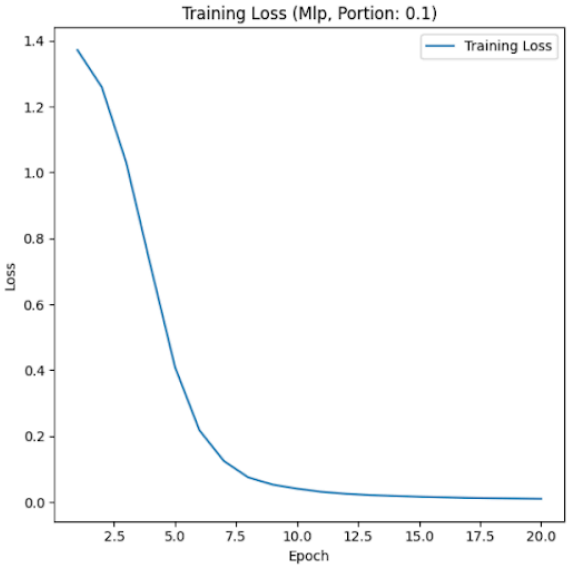

Training Loss (Linear, Portion: 0.5)
Training Loss
Loss
Epoch

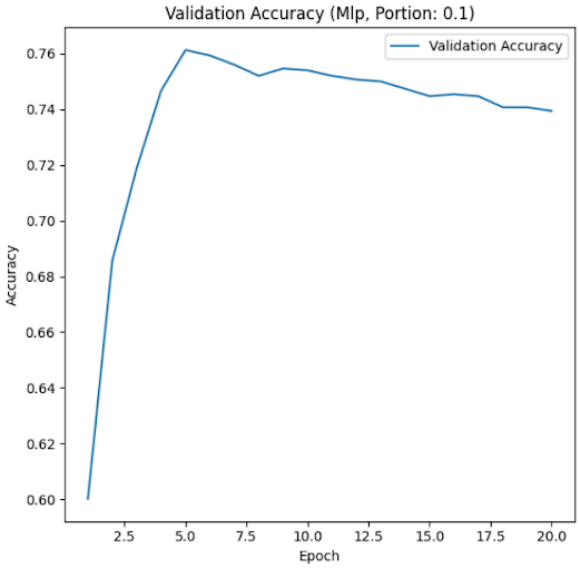

Validation Accuracy (Linear, Portion: 0.5)
Validation Accuracy
Accuracy
Epoch

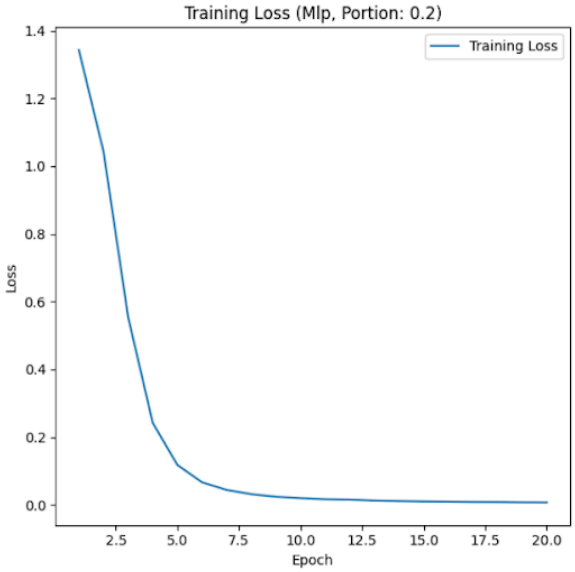

Training Loss (Linear, Portion: 1.0)
Training Loss
Loss
Epoch

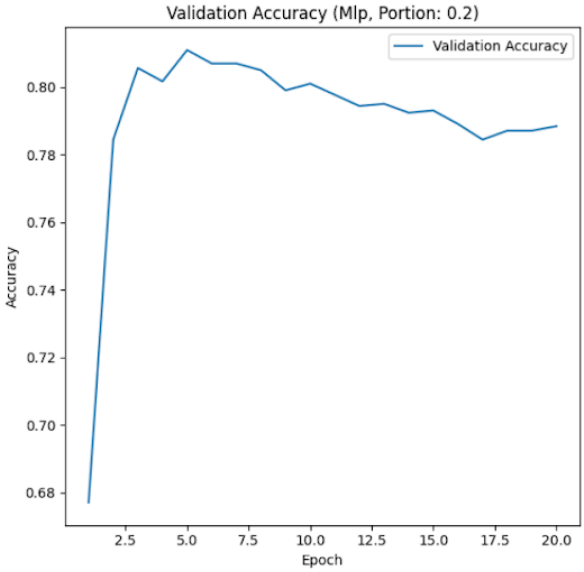

Validation Accuracy (Linear, Portion: 1.0)
Validation Accuracy
Accuracy
Epoch

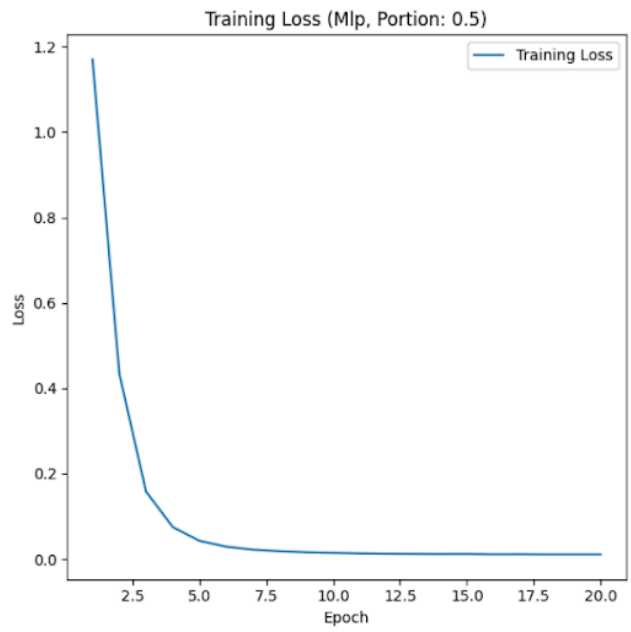
Task 2 :

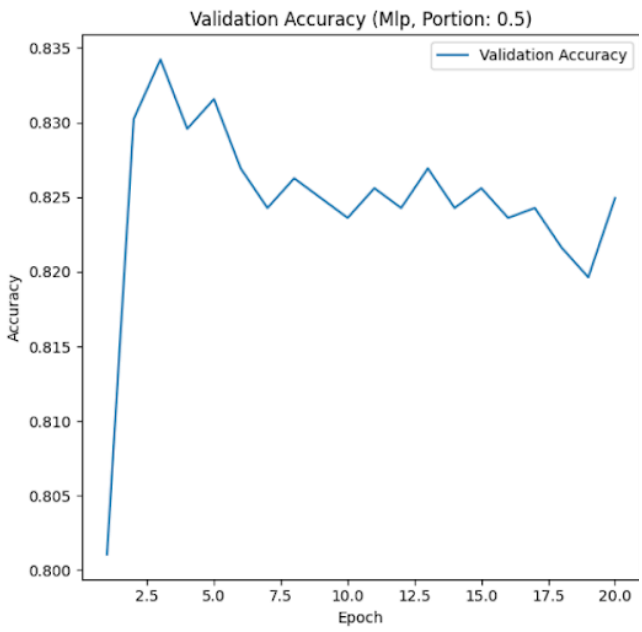

Training Loss (Mlp, Portion: 0.1)
Training Loss
Loss
Epoch

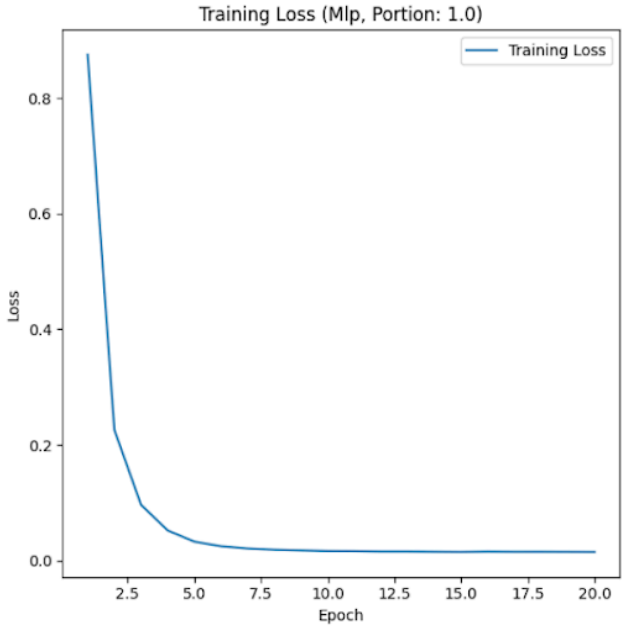

Validation Accuracy (Mlp, Portion: 0.1)
Validation Accuracy
Accuracy
Epoch

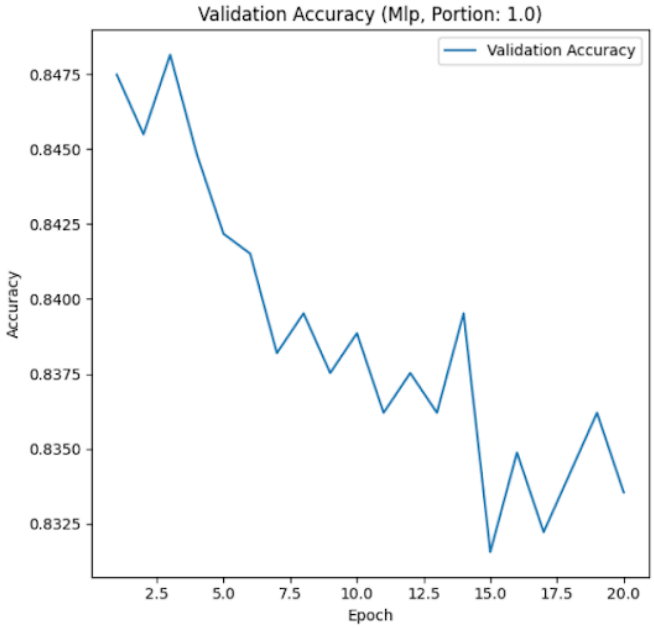

Training Loss (Mlp, Portion: 0.2)
Training Loss
Loss
Epoch


Validation Accuracy (Mlp, Portion: 0.2)
Validation Accuracy
Accuracy
Epoch

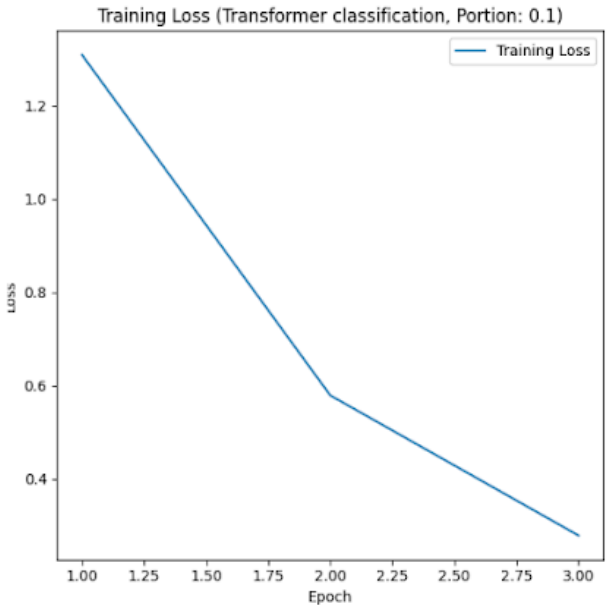

Training Loss (Mlp, Portion: 0.5)
Training Loss
Loss
Epoch
1.2
1.0
0.8
0.6
0.4
0.2
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

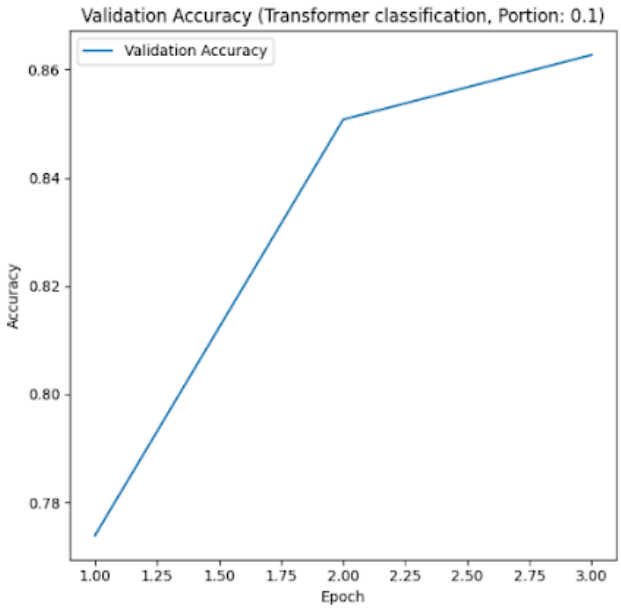

Validation Accuracy (Mlp, Portion: 0.5)
Validation Accuracy
Accuracy
Epoch
0.835
0.830
0.825
0.820
0.815
0.810
0.805
0.800
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

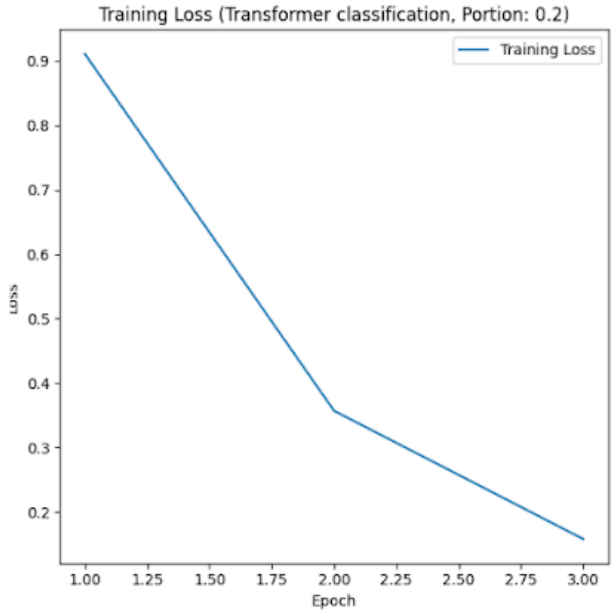

Training Loss (Mlp, Portion: 1.0)
Training Loss
Loss
Epoch
0.8
0.6
0.4
0.2
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

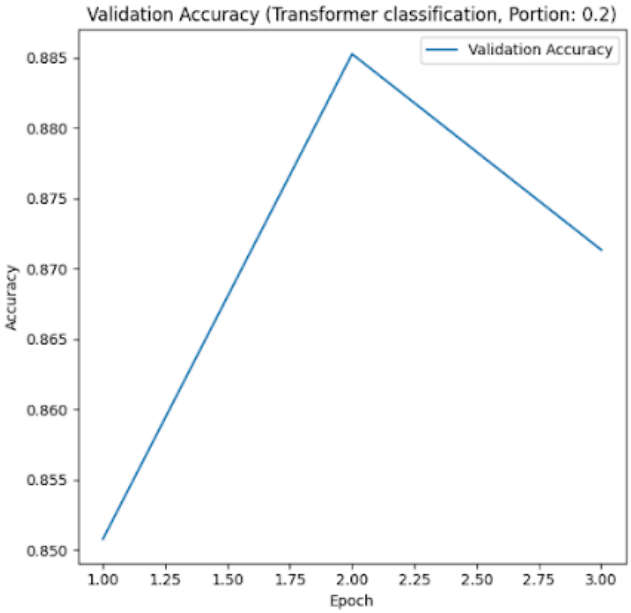

Validation Accuracy (Mlp, Portion: 1.0)
Validation Accuracy
Accuracy
Epoch
0.8475
0.8450
0.8425
0.8400
0.8375
0.8350
0.8325
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

Task 3 :


Training Loss (Transformer classification, Portion: 0.1)
Training Loss
Loss
Epoch


Validation Accuracy (Transformer classification, Portion: 0.1)
Validation Accuracy
Accuracy
Epoch


Training Loss (Transformer classification, Portion: 0.2)
Training Loss
Loss
Epoch

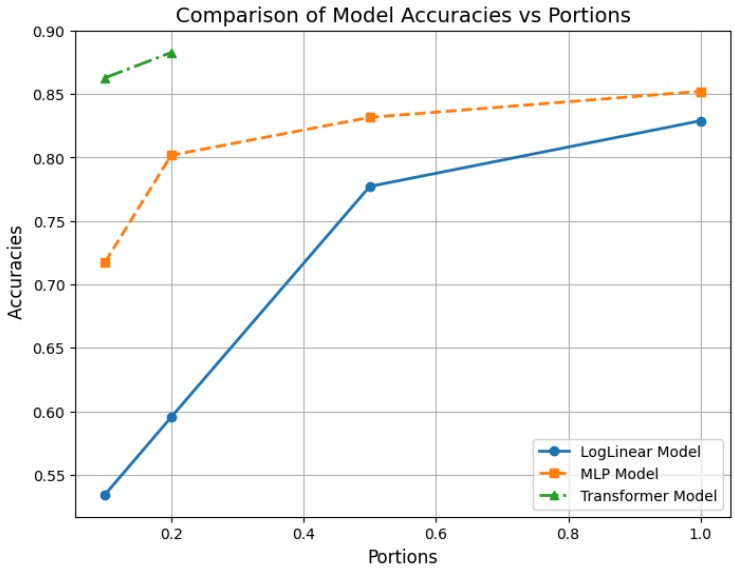

Validation Accuracy (Transformer classification, Portion: 0.2)
Validation Accuracy
Accuracy
Epoch

Task 4 :

(a) Which model had the highest accuracy?

The Transformer model had the highest accuracy reaching above 0.88 when training for 2 epochs and portion 0.1.

(b) Which model was the most sensitive to the size of the training set?



If we restrict ourselves to the first 3 epochs (since all 3 models were trained for at least 3 epochs) we can see that the slope of the accuracy vs portions is the largest for the MLP model, which means the size of the data portion affected the MLP model the most. However, if we look at the overall trajectory of the 3 models, we can see that the LogLinear model has the highest trajectory making it the most sensitive to the size of the training set.

(c) Extract (in your code) and report the number of trainable parameters in each model. Do additional parameters help? Explain.

In the Single Layer Model we have 2000 features which have to be weighted in their TF IDF vector, and each of them needs to have a weight with one of the 4 possible classes. Thus, adding up 2000*4 and the 4 biases for each of the classes gives us a total of 8004 parameters.

In Multi Layer Perceptrons, we have 2000 features, a hidden layer of size 500, an output layer of size 4. In a similar manner to the previous model, we have 2000*500+500*4+ 500 + 4 = 1,002,504

In the Transformer model and as extracted by the following methods: total_params = sum(p.numel() for p in model.parameters()) , the total number of parameters is 82,121,476.

Adding parameters might help the model learn even finer rules and patterns in the training data however this comes with the risk of overfitting the model to the training data. So, if the data isn’t highly representative of the real world we might see the accuracy going down over the test data when parameters’ number goes up.