Introduction To NLP

Assignment 2: Report

Hitesh Goel
2020115003

Results and Analysis:

Hyperparameters used

I trained four different models for various parameters. Here are the parameters used:

Hyperparameters	Model_1	Model_2	Model_3	Model_4
embedding_dim	128	128	256	256
hidden_dim	128	256	128	128
num_layers	2	4	5	6
dropout	0.15	0.15	0.15	0.25
learning_rate	0.01	0.01	0.001	0.001
num_epochs	5	5	10	10

Scores

Dev Set	Accuracy	Precision	Recall	F1 Score
Model_1	87.9	93	95	95.8
Model_2	89	93.5	95	93.2
Model_3	95.1	97.6	96.8	96.7
Model_4	93.3	96.3	96.9	95.8

Test Set	Accuracy	Precision	Recall	F1 Score
Model_1	88.5	93	95.2	93
Model_2	89	93.2	95	93.5
Model_3	95.4	97.9	97.1	97
Model_4	94.3	96.8	97.2	96.3

Analysis

Varying Learning Rate and Number of epochs
We know that for a higher learning rate, the function would converge to a local minimum quicker and hence, only 5 epochs are used. However, a higher learning rate might not be optimal and could give us a minimum which is not ideal. Hence, the learning rate was decreased for a few parameters, and subsequently, the number of epochs was also increased to 10. A jump in accuracy was seen.

Varying embedding size and hidden size
The embedding size and hidden size were varied between 128 and 256, but there were no significant changes seen to the scores because of these hyperparameters.

Varying the number of layers and dropout
This is perhaps the most important parameter. For a very low number of layers (2), we notice that our accuracies are not very good. On increasing the number of layers to 4 and 5, a clear jump is noticed, and the model improves significantly. This is because increasing the layers increases the complexity of the model, and it is able to learn the parameters better.

However, we tried increasing the layers further to 6, with a subsequent increase in the dropout parameter, so as not to overfit, but the accuracy of the model decreased slightly.