1. Describe the details for your best performing model: parameters such as number of epochs, batch size, and learning rate. Compare and contrast with other parameter settings you used—feel free to use graphs or tables to compare. Why do you think it performed better than the other models?

Ans:

For my best performing model, I used the following hyperparameters:

I experimented with different configurations of these parameters to see their impact on performance. Below, I'll describe the different settings I tried and compare the performance of each.

Comparison of Hyperparameter Settings

Parameter Setting	Epochs	Batch Size	Learning Rate	F1 Score	Train Loss
Model 1	100	16	0.1	0.86	0.44
Model 2	100	20	0.01	0.86	0.37
Model 3	100	16	0.005	0.90	0.43
Model 4	200	16	0.05	0.87	0.26

When I increased the number of epochs with decreasing learning rate, I could see that I was getting a better F-1 score for Dev and Test. I could understand that when LR rate is slow there is a slow convergence of the model and I get a better result.

If LR is more, the slow convergence is not there and the values are dynamic and are oscillating throught the loss calculation process.

2. What do you think would happen if you didn't normalize the feature vectors? Write down a guess for what you think would happen, and then run an experiment to test your intuitions and report back what you learned.

Ans:

If I don't normalize the feature vectors, I expect the model to perform poorly because features with larger values could dominate the learning process, making the model biased toward those features. The gradients might also become unbalanced, causing slow or unstable learning. It would have Magnitude differences, convergence issues etc.

After experimenting without normalizing the feature vectors, I could see that my hypothesis was correct. For the best model for which I was getting the F1 score above 90, it significantly dropped.

3. What would happen if you removed one of the features entirely and used a 5-dimensional feature vector? Choose one feature and remove it from your vector. Then, run another experiment and see what happens. Does the test F1 go up or down? Does the model converge slower, or faster? Report which feature you removed and what you learned.

Ans:

If I remove one of the features entirely and use a 5-dimensional feature vector, I expect the model to lose some information, which could lead to a decrease in performance. Removing a feature might make the model converge faster since there are fewer dimensions to learn, but it could also result in lower accuracy or a decrease in the F1 score because the model is missing important data.

I remove avg_word_length as a feature and I could see that the F1 score decreased a little bit. There was a significance difference when I removed positive or negative word count. The model convergence has a negligible speed increase for avg_word_length but model converged faster if I removed the positive and negative features.

4. Review Section 4.10, p. 18 of the textbook and then consider the resources we used for this task: for instance, the training data and the positive and negative lexicons. Did you notice any biases present in these resources? Can you think of any harms or unintended consequences (harmful or not) that this classifier could cause? There is no correct answer; just write a couple of sentences reflecting on this prompt.

Ans:

I noticed that there might be some biases in the positive and negative lexicons. For example, the negative lexicon appears to contain a broader variety of words, some of which may be mislabelled, leading to inaccurate predictions. This could result in unfair or skewed classifications, especially if certain negative words that are not actually related to reviews are included. One potential harm of this classifier is that it could incorrectly label neutral or complex statements as negative, influencing sentiment analysis in ways that could be misleading for users relying on the system for feedback or decision-making.