Natural Language Processing - IMDB Movie Review							
	Description	Hyperparameters	Number of Epochs	Training Loss	Training Accuracy	Test Accuracy	Comments
Part 1a	Given model - Word Embedding Layer + Fully Connected Layer + Batch Norm Layer + Relu + Dropout + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=500	6	0.1421	94.55%	87.06%	Describe more about the model/results such as why certain hyperparamters were chosen or the effect it had on the accuracy/training time/overfitting/etc.
	Given model - Removed Dropout	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=5000	6	0.0844	97.09%	79.65%	Just by removing the dropout layer and increasing the hidden layer size by an order of magnitude the model's overfitting has gotten worse. Run time increases by about 90%.
	Given model - Removed Dropout	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=50, HiddenUnits=8000	6	0.6184	67.15%	63.66%	The model's accuracy greatly decreases with such a small vocabulary size.
	Given model	SGD optimizer with LR=0.1 reduced every 10 epochs by a factor of 10, BatchSize=200, VocabularySize=8000, HiddenUnits=500	30	0.2152	91.34%	82.37%	SGD requires more epochs, but still doesn't achieve same accuracy.
Part 1b	Given Model - Fully Connected Layer + Batch Norm + Relu + Dropout + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=500	6	0.3008	87.49%	84.96%	Time for each epoch is much shorter than embedding withough GloVe features.
	Given Model - Fully Connected Layer + Batch Norm + Relu + Dropout + Output Layer	SGD optimizer with LR=0.1 reduced every 10 epochs by a factor of 10, BatchSize=200, VocabularySize=100000, HiddenUnits=500	15	0.2899	88.00%	81.39%	Even with more epochs SGD doesn't reach Adam accuracy.
	Given model - Removed Dropout	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=5000	15	0.1132	95.82%	84.42%	Just by removing the dropout layer and increasing the hidden layer size by an order of magnitude the model's overfitting has gotten worse.
	Given model - Removed Dropout	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100, HiddenUnits=500	6	0.5678	70.89%	66.12%	The model's accuracy greatly decreases with such a small vocabulary size.
Part 2a	Given model - Word Embedding Layer + LSTM + Batch Norm Layer + Dropout + Max Pool + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=500	20	0.0825	97.10%	87.36%	
	Given model - Word Embedding Layer + (LSTM + Batch Norm Layer + Dropout) x 2 + Max Pool + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=500	20	0.0729	97.50%	86.48%	Adding the second LSTM layer increases runtime by 50%, but with a slight decrease in accuracy.
	Given model - Removed Dropout	ADAM optimizer with LR=0.001, BatchSize=20, VocabularySize=8000, HiddenUnits=500	20	0.1069	96.14%	87.26%	Decreasing batch size reduced accuracy but increased run time by 250%.
	Given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=50	20	0.2836	88.12%	86.12%	Decreases training time by 66% with a small decrease in test accuracy. Less overfit.
Part 2a	Given model - LSTM + Batch Norm Layer + Dropout + Max Pool + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=500	20	0.2136	91.27%	90.04%	

	Given model - (LSTM + Batch Norm Layer + Dropout) x 2 + Max Pool + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=500	20	0.1997	92.08%	90.45%	Increases training time by 100% with a small increase in accuracy.
	Given model - LSTM + Batch Norm Layer + Dropout + Max Pool + Output Layer	ADAM optimizer with LR=0.001, BatchSize=20, VocabularySize=100000, HiddenUnits=500	20	0.1554	94.08%	90.32%	Decreasing batch size increased training time by 300% with slight drop in accuracy.
	Given model - LSTM + Batch Norm Layer + Dropout + Max Pool + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=50	20	0.3081	86.89%	88.88%	Decreased hidden layer dimension reduced training time by 50%, but the model is now underfit.