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 + Mean pooling + Fully Connected Layer +Batch Normalize layer +Relu + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=500	6	0.1422	94.38%	87.20%	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.
	Custom 1 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=1000	6	0.1209	95.37%	86.11%	The number of hidden units increased to 1000 which made the model overfit. The training accuracy increases a little bit, but the test accuracy is decreased by 1%. Besides, increasing the number of hidden units increases the training time greatly, from 15 seconds to 26 seconds per epoch.
	Custom 2 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=10	6	0.4314	81.55%	83.22%	The number of hidden units decreased to 10 which made the model underfit. The test accuracy and training accuracy decrease a lot and the training loss is still high, meaning that the model is underfitting.
	Custom 3 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=800	6	0.1397	95.06%	86.98%	The number of hidden units increased to 800 which made the model a little bit overfit. The test accuracy just decrease a little.
Part 1b	Given Model - Fully Connected Layer + Batch Normalize layer + Relu + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=500	6	0.2961	87.66%	85.86%	The model doesn't seem to suffer from overfitting as the training accuracy and test accuracy are quite close to each other.
	Custom 1 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=5000	6	0.2901	88.11%	84.61%	The number of hidden units increased to 1000 which made the model overfit. The training accuracy increases a little bit, but the test accuracy is decreased by 1%. The training time this time didn't change too much.
	Custom 2 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=10	6	0.3702	83.96%	85.46%	The number of hidden units decreased to 10 which made the model underfit. The test accuracy and training accuracy decrease a lot and the training loss is still high, meaning that the model is underfitting.
	Custom 3 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=800	6	0.2937	88.06%	85.36%	The number of hidden units increased to 800 which made the model a little bit overfit. The test accuracy just decrease a little.

Part 2a	Given Model - Word Embedding Layer + LSTM Layer + Batch Normalize Layer + Locked Dropout + Max Pooling + Fuly Connected Layer + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=500, SequenceLengthOfTrain=100, SequenceLengthOfTest=400	20	0.1015	96.27%	87.71%	The model does not perform very well because enven though the training accuracy reaches 96.27%, the test accuracy is just 87.71%.
	Custom 1 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=500, SequenceLengthOfTrain=400, SequenceLengthOfTest=400	20	0.0166	99.40%	79.00%	The sequence length for training is increased from 100 to 400. The model is now overfitting. Training accuracy is approaching 100% while test accuracy is only 79%, which is worse than the given model.
	Custom 2 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=8000, HiddenUnits=100, SequenceLengthOfTrain=100, SequenceLengthOfTest=400	20	0.2243	90.92%	84.51%	The number of hidden units is decreased to 100 to make the model underfit. Now the training accuracy is smaller than the given model. Interestingly, the test accuracy doesn't decrease much. A potential explanation is that the given model is overfitting so decreasing the number of hidden units is actually alleviating this problem.
	Custom 3						
Part 2b	Given Model - LSTM Layer + Batch Normalize Layer + Locked Dropout + Max Pooling + Fuly Connected Layer + Output Layer	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=500, SequenceLengthOfTrain=100, SequenceLengthOfTest=400	20	0.2104	93.50%	91.02%	The model doesn't seem to suffer from overfitting as the training accuracy and test accuracy are quite close to each other. However, it might be a little bit underfitting because during the training process, the test accuracy is still increasing.
	Custom 1 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=500, SequenceLengthOfTrain=400, SequenceLengthOfTest=400	20	0.0707	97.45%	82.58%	The sequence length for training increased from 100 to 400. The model is now overfitting. Training accuracy is approaching 100% while test accuracy is only 82.58%, which is worse than the given model.
	Custom 2 - Same as the given model	ADAM optimizer with LR=0.001, BatchSize=200, VocabularySize=100000, HiddenUnits=100, SequenceLengthOfTrain=100, SequenceLengthOfTest=400	20	0.3005	86.98%	86.23%	The number of hidden units decreased to 100 making the model underfit. Now the training accuracy is smaller than the given model. Interestingly, the test accuracy doesn't decrease much. A potential explanation is that the given model is overfitting so decreasing the number of hidden units is actually alleviating this problem.
	Custom 3						