

Question and Answers:

1. What made you decide on text generation?
 - a. I thought the mechanics behind it were very interesting. I also wanted to see how different source inputs resulted in outputted text that reflected the character of the input text.
2. Why Sherlock Holmes?
 - a. I had previously done a similar LSTM text generator using the text of the Hobbit and I wanted to do something a little different. I wanted to use public domain texts, and Sherlock Holmes, like the works of JRR Tolkien, are instantly recognizable. You can tell from the text that the adventures of Mr. Holmes were involved and not just because we kept talking about Watson.
3. What was the reasoning behind the different temperatures you used?
 - a. I wanted a broad spectrum and I also wanted some key values in there. As humans, we like order, so by including .25, .75, and 1.25 in addition to the .1 increments we can use those values as sort of anchor points. I also knew that our lower values were more likely to be caught by our classification and I wanted to show a good spread over time.
4. Are there any examples of games that actually use similar technology?
 - a. I had heard of it vaguely before I began, but AI Dungeon (<https://play.aidungeon.io/main/landing>) is basically this concept in working game form. The adventure is procedurally generated by an AI and is fascinating!
5. What was the most surprising thing you discovered?
 - a. I was shocked at how well the logistic regression was able to pick out the generated text. There also was not as nearly as big a gap between the mid-level temperatures and the high temperatures as I expected. For a human, the 1.25 temperature text is easy to pick out, but the algorithm was not much worse at the .7 temperature.
6. What refinements to the process do you see being the next steps?
 - a. Our text generation algorithm has some room to grow. Not only can we tweak some of the inputs, but we could also vary the number of epochs for training. We might even bring in other text generation algorithms. We could weight each model differently and choose the next character by consensus.
7. What causes the random nonwords in the generated text?
 - a. The LSTM Text Generation method we use actually generates the probabilities for the next character in our sequence, not word. Thus, it is evaluating the odds of the next character being an e, not the odds of the next word being elementary. This is why we occasionally have nonwords generated.
8. You mentioned generating characters instead of words. Would generating text result in better generated text?
 - a. Most likely. At least from the user standpoint, at least the actual words would make it less obvious that it was generated text. If you have ever participated in an iPhone text prediction meme, you will know that when you predict the next words it is easy to end up in circles repeating the same few words over and over again.
9. Why did you choose logistic regression for the classification algorithm?
 - a. In a previous project, a spam classifier, we found that between the couple of models that we utilized the Logistic Regression seemed to be the most accurate. As we had

experience implementing that particular algorithm and it behaved the best, it was a solid choice for this project as well.

10. Isn't the F-Score the same as the accuracy? Why were both reported?

- a. While the F-Score is similar to the accuracy, it actually incorporates more than one metric in a single quantifiable number. In this case, while the F-Score is also very close the accuracy for a few of the datasets, it does differ slightly as it also incorporates a measure of the precision of the model.