

Shakespeare Sentiment Analysis

Lindsey Vanderlyn and Devynn Diggins

Project Overview: For our project we decided to tackle the idea of sentiment analysis within some of the plays of William Shakespeare. We downloaded the complete works of Shakespeare from Project Gutenberg, separated it into plays and each play into acts, and then performed sentiment analysis for important characters in the play, keeping track of how their positivity and negativity changed from act to act and comparing the overall sentiment of characters throughout the play to that of characters in other .

Implementation: Our code is situated in two programs, Playfinder2.py and ShakespeareSpeech.py. Playfinder searches our text, Shakespeare.txt, for specific plays compatible with our code, and returns them as a list consisting of a dictionary and a series of strings. The dictionary contains all of the names of the characters in the play's Dramatis Personae, and each of the strings contains an entire act of the play.

ShakespeareSpeech contains a series of functions: In this section of code, we make use of the sections of text broken up by Playfinder2. The general flow of processing happens by breaking the string text up into a list of all the words in the play using regular expressions and then searches through this list matching lines each character says to keys in the dictionary determined by the dramatis personae in playfinder. After the lines have been broke up correctly, we return the dictionary values to strings, broken up by punctuation and use pattern to run sentiment analysis on these lines. After we have these values we then can compare the general sentiment of characters within and across plays as well as how the sentiment of a single character develops throughtout a play. We then wrapped all of these functions up with 'soul' which allows the user to choose what they wish to do with the function

Our functions use Regular Expressions and Matplot Lib to function. Please use these or you will have a bad time!

Results: We found that overall, sentiment analysis for any given character tended to the neutral side of things, which is expected, because the sentiment is calculated through running every spoken line of each character and calculating an average sentiment from that. We also found that there were large similarites between many of Shakespeare's characters, likely as a result of the above.

Reflection: This was a good partnership. Although our initial ambitions were perhaps slightly too large for the time we had, I feel that we made good use of our time and were able to present a decent graphic analysis of the similarities between characters although we ran out of time for incorporating the quantative analysis we had originally hoped for. Over all we felt that we had a good expereinece where we learned a lot both about interacting with git hub and about exploring different applications for programming/practicing programming skills. The level of engagement for this project was fairly equal between both parnters which led to a smooth partnership and greater respect for eachother as collaborators and individuals. In terms of happiness I feel that we had a good experience

and enjoyed working with each other and the problem in order to come up with an answer (although sometime debugging made us feel like idiots.)