I tried to analyze Charles Dickens's literature from different time period of his life to see the shift in his writing style and dominating emotion state in his books.

To implement my analysis, I first imported the text from the online database to my program. Then I stripped the preamble and various term&conditions in the text with string slicing. I could have used a function to do this, but due to the various spacing/content difference of such things. I chose to deal with each of them separately. Then to avoid download data from the server every time I run the code, I stored all the text as a string locally.

After getting the raw data, I started my analysis with word frequency analysis. I wrote a dictionary to output the top ten most common words in each text. Also, I used the sentiment function from Pattern library to analyze the general emotion of each text.

For the most frequent word test, I didn't get the satisfactory outcome as expected. When I try to generate the top ten/twenty or even thirty words in the text, all I got were some fundamental words of the English language, such as 'am', 'for', 'in' and etc. For the sentiment test, however, I did get the data I want. Dickens's five novels, “Oliver Twist”, “David Copperfield”, “Bleak House”, “A Tale of Two Cities” and “Great Expectations” , have the following result: (0.086, 0.493), (0.113, 0.503), (0.100, 0.499), (0.079, 0.491) and (0.079, 0.491). We can see that Dickens is a pretty pessimistic person. His books in his early days and late years are full of negative words. His work in his middle ages tend to be a little more positive, but only to a limited extents.

When looked back this project, I realized that I should have thought earlier what the possible outcome I'd get from my program and that could probably prevent the “fruitless” word frequency analysis.