Text Mining

(using Tries for sorting ‘Big Data’)

By David Alves & Elitsa Marinovska

WORK DESCRIPTION

Our group would like to implement a Tries data structure that can be applied using the English alphabet. This data structure will serve the purpose of sorting all the words used in the novel “A Tale of Two Cities” by Charles Dickens, “Complete Works of William Shakespeare” by the titled author and our course study book named “Algorithms, 4th Edition” by Robert Sedgewick and Kevin Wayne. We would be interested in finding out how many times a word occurred in the previously mentioned written works. The data structure should be addressed with the mindset of designing it in a generic way to operate with strings, therefore not putting any additional requirements on the big data we want to sort apart from being an English text. The implementation of the Tries will be managed in Java, obeying all the standard principles associated with it and trying to put into use the best coding practices. The group decided to export the results into several CSV files, so graph visualization can take place through Microsoft Excel and conclusions can be drawn from those.

REPORT STRUCTURE

Contributions

Before heading towards finding a solution on the above-mentioned matter, the team conducted a research on published statistics about the words with the highest overall frequency in the English language. The resource we stumbled upon was about the “Most common words in English” article in Wikipedia, more specifically the section about the 100 most common words in English. This page had also resource links pointing to very interesting studies by the Oxford English Corpus: “Facts about the language” [1]. Additionally, worth mentioning is the fascinating theory about Zipf's law [2], which is a theory stating that the frequency of any word is inversely proportional to its rank in a frequency table.

---

[1] For an overview of the studies you can visit the following page - https://web.archive.org/web/20111226085859/http://oxforddictionaries.com/words/the-oec-facts-about-the-language

[2] Zipf’s law - https://en.wikipedia.org/wiki/Zipf%27s_law

Background

Having the list with the most statistically used words in English, we wanted to compare this results with the ones that our program would point out regarding well-known texts from previous centuries. Additionally, the presence of the Algorithms book could shine some light on how the word usage has evolved over the year, as well as which words are typically used when creating a book with a more scientific perspective. With the previously obtained knowledge about the Zipf's law, we would be interested into finding out how much would our results be in accordance with its statements.

Research questions

To summarize the statements above, we would like to put our focus into creating graphs that would represent the word usage from different time periods and how would the three chosen book sources compare to Zipf's law.

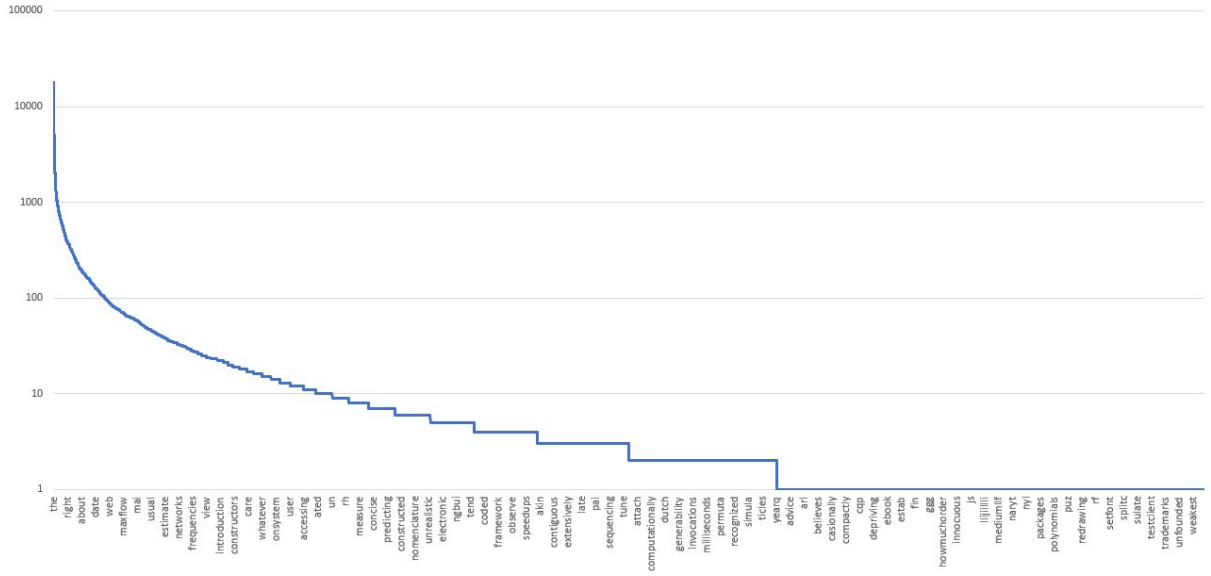
Methodologies

The chosen methodology to analyse the texts would be the Trie data structure. The team had previously experience with sorting and counting words from relatively big data sources, and due to the time taken by those we were very eager to explore the inner works and strengths of the new concept. This type of structure can also be used in other different scenarios apart from strings, but in the present case we decided to test its performance.

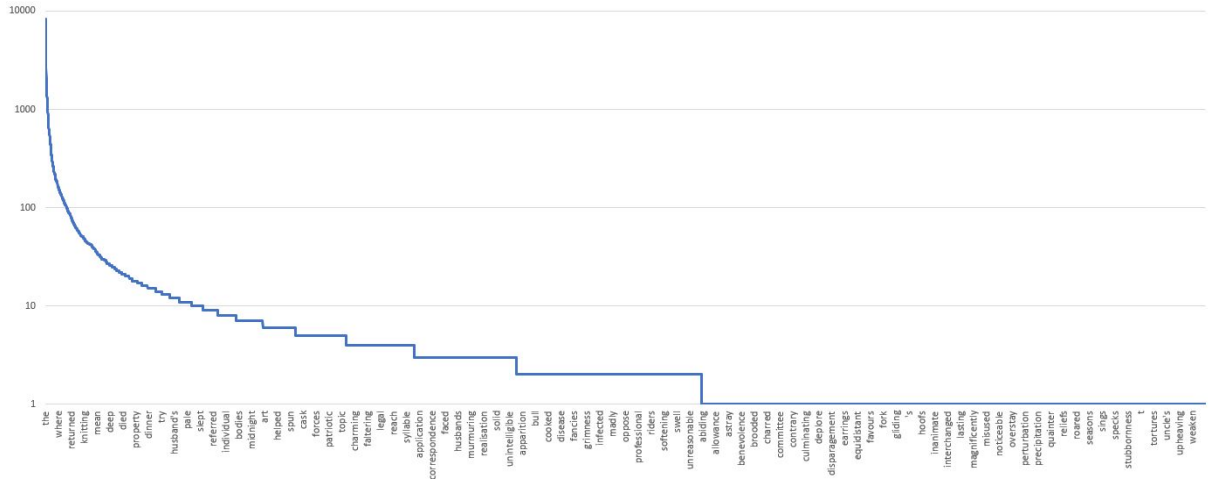
Findings

The reached conclusion for the three text pieces is that the type of words that occur in their top 10 rank are one syllable words serving as articles, preposition and conjunctions. As expected, this was also the scenario with the most common English words. We have used the data outputted in the CSV files to generate the graphs below, which provides a better visualization of our findings.
Referencing back to the Zipf's law, we noticed that even though it is not a concrete science (⅔ of the text didn't have a decrease of half of the count between 1st and 2nd rank) but even so, when all the results were plotted in a graph we could see the presence of a Zipfian distribution.

Algorithms



Tale of Two Cities



Shakespeare