Design and Analysis of Algorithms

Manuel Lopez
100622700

Simple plagiarism detection: Enhance Levenshtein's Distance

The levenshtein distance between two documents find the minimum number of edits to transform the first document to the second. Levenshtein distance only allows for three types of edits Insert, Deletion, Substitution. To improve the levenshtein distance we preprocess the document to remove all the stop words and punctuation. Most search engine remove stop words because they give no real meaning to the algorithm. Levenshtein distance complexity is O(n1 * n2).

How program levenshtein algorithm works:

Set **A** = length of document Original
Set **B** = length of document of Copy
Initialize a 2d array row size of **A** and Column size of **B** called distance
Observe the value of d2 (i from 1 to a).
Observe the value of d1 (j from 1 to b).
If the value at d2[i] is equals to value at d1[j], the cost becomes 0.
If the value at d2[i] does not equal d1[j], the cost becomes 2.
Set block of matrix Distance[d1, d2] of the matrix equal to the minimum of:

- the block immediately above add 1: Distance [d2–1, d1]
- the block immediately to the left add 1: Distance[d2, d1–1] +1.
- The block is diagonally above and to the left adds the cost: Distance[d2–1, d1–1] + cost.

To get similarity ratio:
((length(Document1) + length(Document2) - distance[row][col]) / (length(Document1) + length(Document2)))
O(n1*n2)

To preprocess Documents:

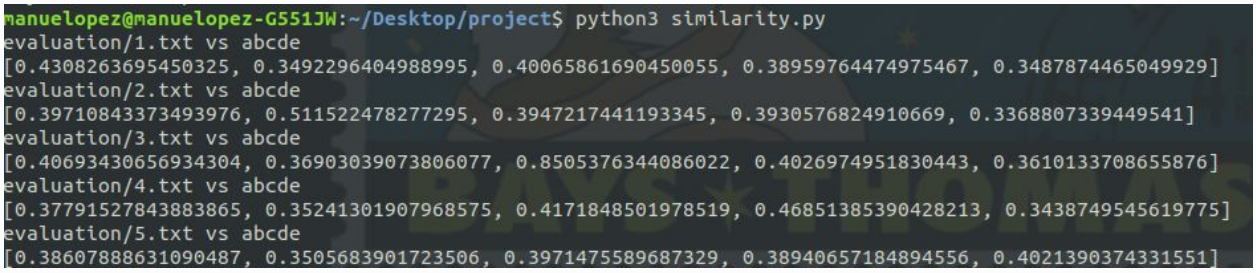
Memory complexity O(n*n): All files are open as read byte then decoded to fix problems with utf-8 characters for comparison by pair then saved to an array.
We loop through the array removing all the Stop words and punctuation.
To remove punctuation I used the String library

After Documents are preprocessed we send the originalProcessed string and copyOriginal string to the levDistance function and get the similarity ratio between the two strings.

Loop through all the Documents while getting the ratio Save each ratio and then saving it to a csv File.

To run program:

levDistance folder contains **similarity.py** open terminal or command prompt and run Python3 similarity.py. All values are saved into a csv file called csv_out.csv



References:

https://link.springer.com/chapter/10.1007/978-981-13-0755-3_6