Code Similarity Project Report

Our goal in this project is to find out the similarities between the code bases written for the previous deidentification project. In other words, we want to see how syntactically close each code base is to every other code base. There are different approaches to address this problem. One possible approach is to look at each code base as a written text and find out how similar different code bases are. An interesting scheme is the Universal Sentence Encoder in TensorFlow Hub that measures sentence similarity and classification tasks.

To measure the similarity of codes, first of all , I have read the data and collected the codes that each student has written to be able to compare different source codes. In the preprocessing step, I have removed the general input-output comments that tends to be highly similar for different codes, since the we all started to write the code based on the first source code that was shared with us in the class. Therefore, I have removed the comments within the triple quotation marks. Moreover, I have replaced the underline symbol with space so that the names make more sense for the universal sentence encoder to capture the similarities with higher chance.
I have implemented the code for universal sentence encoder similar to the universal sentence encoder colab and added the final results as "**Similarity1.ipynb**" file to my github repository "*https://github.com/Paarisaa/Code-Similarity-Measures* ".

Another interesting approach to check the similarities is to use Levenstein different metrics that calculates the the number of substitutions and deletions needed in order to transform one string into another one. I have tried to calculate the Levenshtein distance, Jaro similarity, Jaro-Winkler, and ration similarity metrics for the first three codes and provided the results as "**Similarity2.ipynb**" to my github repository "*https://github.com/Paarisaa/Code-Similarity-Measures*". This time I could not install the python Levenshtein library on my laptop and used Google Cloud's online ipython notebook instead.