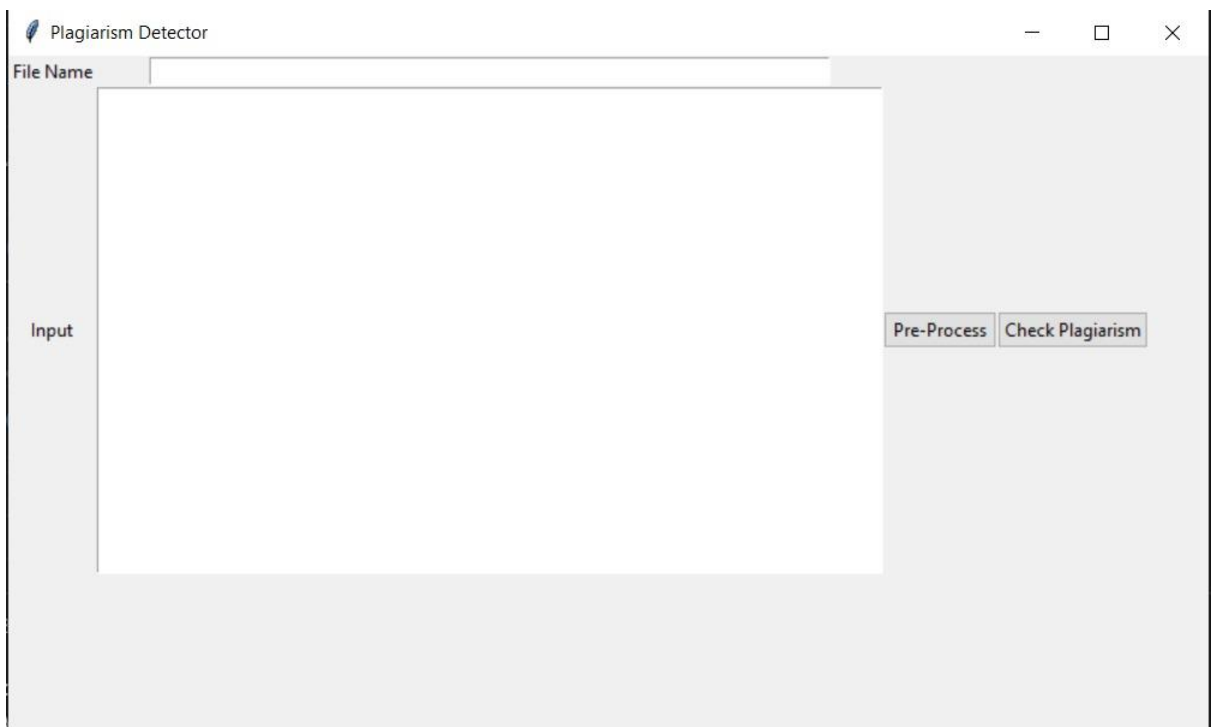


ReadMe

Step 1: Corpus against which we need to check the similarity of the input document has to be loaded into main.py folder first and then run.

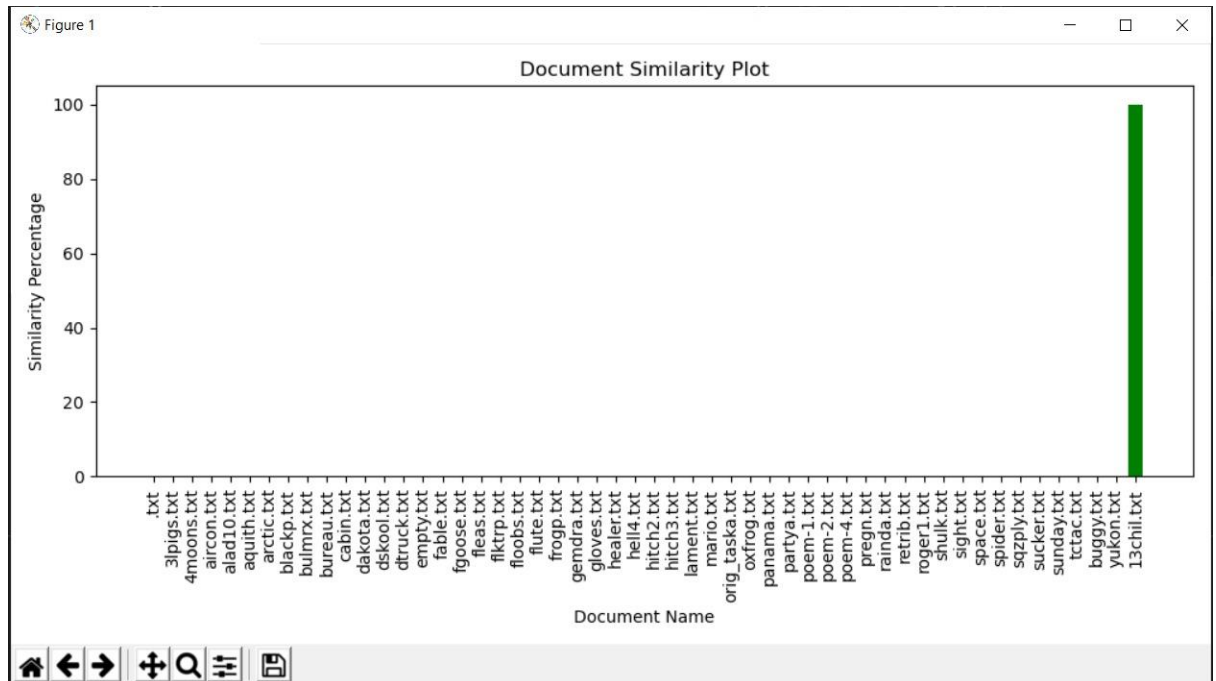
Step 2: We have got an input document, now we have to copy the whole text of the document in the text box shown below. On top of this text box we have to give filename. This text will also be added into the corpus with the filename we have given on top. Now press the pre-process button which will convert documents to trigrams.



The screenshot shows a window titled "Plagiarism Detector" with standard window controls (minimize, maximize, close) in the top right corner. The interface is divided into several sections:

- File Name:** A text input field at the top left for specifying the filename.
- Input:** A large, empty text area in the center for pasting the document content.
- Buttons:** Two buttons are located in the bottom right corner: "Pre-Process" and "Check Plagiarism".

Step 3: By clicking check button we can generate a graph which will give us the extent of plagiarism with each document in corpus.



Packages and Interfaces Needed

1. NLTK : Package used for pre-processing to tokenize and etc.
2. Matplotlib: Plotting library used for drawing the graph
3. TkInter: Python Interface for GUI, helped for making the GUI where the user enters input data and the data gets processed.

Advantages

1. The code works with efficiency if the data or text is copied completely from any source of our corpus. As we can see in our above graph how our code detects 100% plagiarism effectively.
2. Jumbled sentences can also be detected by our plagiarism checker because of trigram method.

Limitation

1. The code lacks when our data in our corpus is heavily paraphrased.
2. It also fails to detect plagiarism if the words are replaced by its synonyms.