A Mini Project Synopsis on

Plagiarism Detection System

T.E. - I.T Engineering

Submitted By

AbhayPratap Singh 19104037

Aditya Joshi 19104044

Bharat Singh 19104043

Under The Guidance Of

Prof. Rajashri Chaudhari

DEPARTMENT OF INFORMATION TECHNOLOGY

A.P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W), Mumbai-400615

UNIVERSITY OF MUMBAI

Academic year : 2021-22

CERTIFICATE

This to certify that the Mini Project report on **Plagiarism Detection System** has been submitted by AbhayPratap Singh(19104037), Aditya Joshi(19104044),Bharat Singh(19104043) and who are a Bonafede students of A. P. Shah Institute of Technology, Thane, Mumbai, as a partial fulfilment of the requirement for the degree in **Information Technology**, during the academic year **2021-2022** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Prof. Rajashri Chaudhari
Guide

Prof. Kiran Deshpande
Head Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.
2.

Place: A.P. Shah Institute of Technology, Thane
Date:

Acknowledgment

This project would not have come to fruition without the invaluable help of our guide **Prof. Rajashri Chaudhari**. Expressing gratitude towards our HoD, **Prof. Kiran Deshpande**, and the Department of Information Technology for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our teacher **Prof. Nahid Shaikh** who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

TABLE OF CONTENTS

Chapter 1

Introduction

Plagiarism in the sense of "theft of intellectual property" has been around for as long as humans have produced work of art and research. However, easy access to the Web, large databases, and telecommunication in general, has turned plagiarism into a serious problem for publishers, researchers and educational institutions. In this paper, we concentrate on textual plagiarism (as opposed to plagiarism in music, paintings, pictures, maps, technical drawings, etc.). We first discuss the complex general setting, then report on some results of plagiarism detection software and finally draw attention to the fact that any serious investigation in plagiarism turns up rather unexpected side-effects. We believe that this paper is of value to all researchers, educators and students and should be considered as seminal work that hopefully will encourage many still deeper investigations.

1.1 Purpose

A plagiarism checker uses advanced database software to scan for matches between your text and existing texts. There are also commercial plagiarism checkers you can use to check your own work before submitting. Behind the scenes, plagiarism checkers crawl web content and index it, scanning your text for similarities against a database of existing content on the internet. Exact matches are highlighted using keyword analysis. Some checkers can also identify non-exact matches (paraphrasing plagiarism). On the user end, the checker typically provides you with a plagiarism percentage, highlights the plagiarism, and lists the sources.

1.2 Objectives

- To avoid plagiarism by ensuring the proper citing of information from outside sources.
- To help researchers know how to use sources ethically and effectively through summarizing, paraphrasing and quoting.
- To help researchers understand how to make use of information from outside sources while maintaining a strong voice of their own.

1.3 Scope

This application can be used to read author manuscripts and student papers in a few minutes, matching what they have submitted to already published work. The application highlights the parts that match/are similar to already existing files/documents. The text that is highlighted includes content that is identified as too similar to published content. The highlighted text is used to generate a score that represents the percentage of similarity between the author's document and published content.

Chapter 2

Project Definition

Plagiarism in the sense of "theft of intellectual property" has been around for as long as humans have produced work of art and research. However, easy access to the Web, large databases, and telecommunication in general, has turned plagiarism into a serious problem for publishers, researchers and educational institutions. In this paper, we concentrate on textual plagiarism (as opposed to plagiarism in music, paintings, pictures, maps, technical drawings, etc.). We first discuss the complex general setting, then report on some results of plagiarism detection software and finally draw attention to the fact that any serious investigation in plagiarism turns up rather unexpected side-effects. We believe that this paper is of value to all researchers, educators and students and should be considered as seminal work that hopefully will encourage many still deeper investigations.

Chapter 3

Proposed System

The proposed system is an application where one can check for plagiarism. It is a software that allows you to check for any kind of plagiarism either by typing out the content that you want to check or upload a file of a supported format to check for plagiarism in that file.

This app will help users to check plagiarism in their content which is an important step for making proper citations and references to the sources from which they have taken the content from. It also highlights the copied content and also shows from which file the content has been taken so the user can properly cite the sources without having to worry about having their research papers rejected.

3.1 Features and Functionality

- **Reset Module :** This module will help user to reset all the content that user has typed aur pasted in the text area with this user can easily do work effectively.
- **Text Module :** An area which is used to type in the paragraph or the text on which plagiarism is to be detected. Here the user can either type in the contents or can copy and paste the parts on which the plagiarism detection is to be done.
- **File Upload Module :** User can also upload their files in txt format for detecting the plagiarism in their content.
- **Submit Module :** When the user has entered the text in the text area or after uploading the file to check for plagiarism he can use the submit button to get the

results related on how much and how much percentage of the content is plagiarized.

- **Result Module :** At the end user can see the percentage of the plagiarized content with respect to the dataset and can the plagiarized content will be highlighted.
- **About Module :** Credits given to the students who have helped us in making this project.
- **Letter counter Module :** This will keep a count of the number of letters that have been typed by the user in the text area and there is a limit of 5000 letters at one time so he can know where to stop.

Chapter 4

Project Outcome:

Plagiarism is very important in our day to day work. With most of the data available to us in digital format the venues for plagiarism is opening up. To avoid this kind of cheating and to acknowledge the originality of the author new detection techniques are to be created. To protect the intellectual property source code new techniques are to be developed and implemented.So it help us to detect the things anyone copied so because of this no one can send any other project.

Chapter 5

Software Requirements

- Windows 7 or above
- Python 3 or above versions
- Tkinter

Python

Python is an interpreted high-level general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

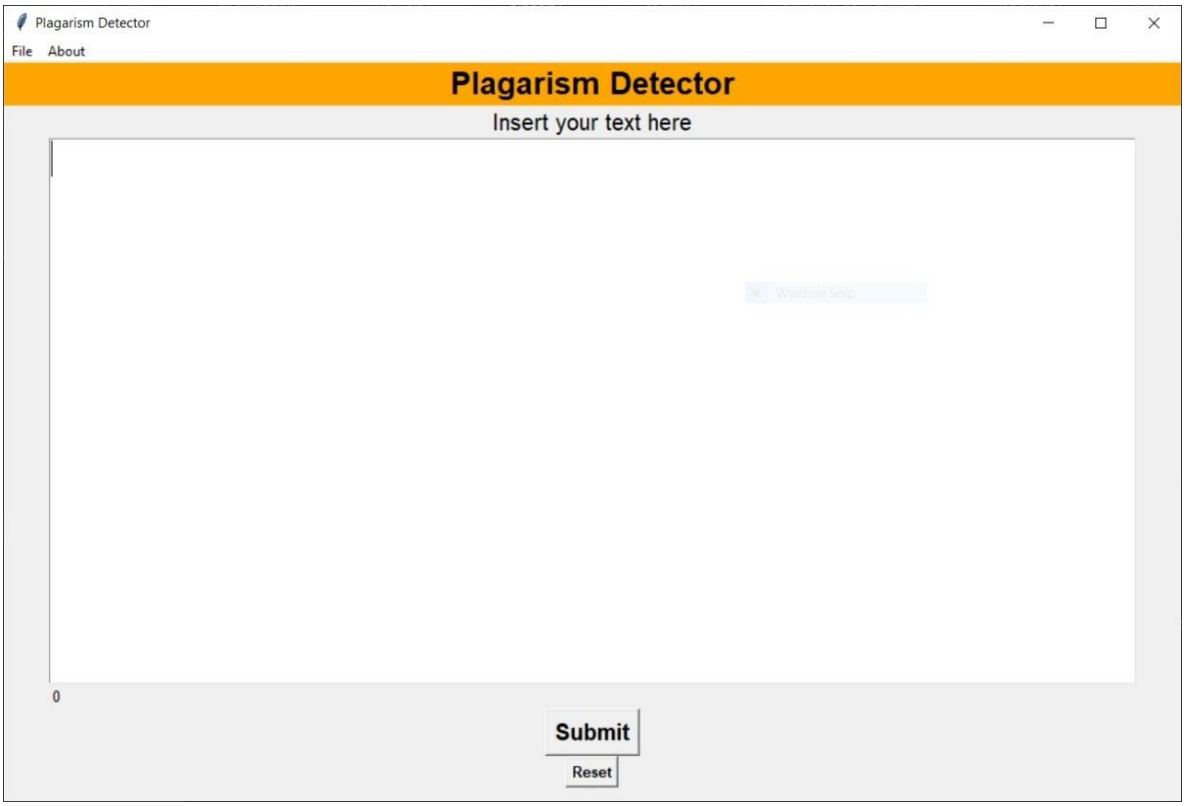
Tkinter

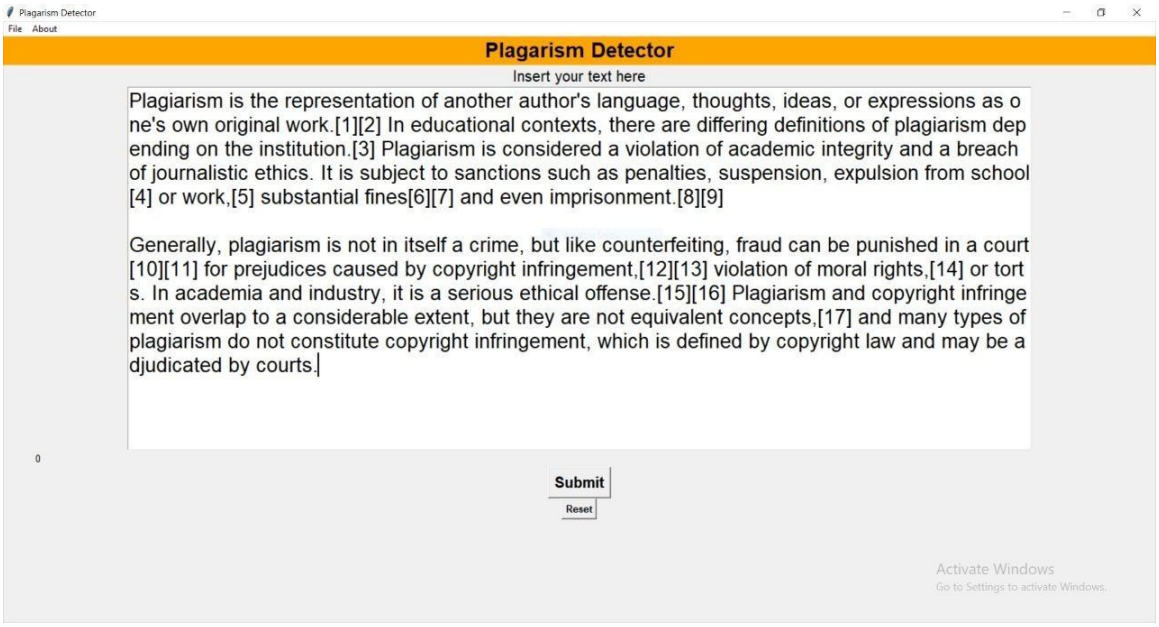
Tkinter is a standard library in python used for creating Graphical User Interface (GUI) for Desktop Applications.

With the help of Tkinter developing desktop applications is not a tough task. The primary GUI toolkit we will be using is Tk, which is Python's default GUI library.

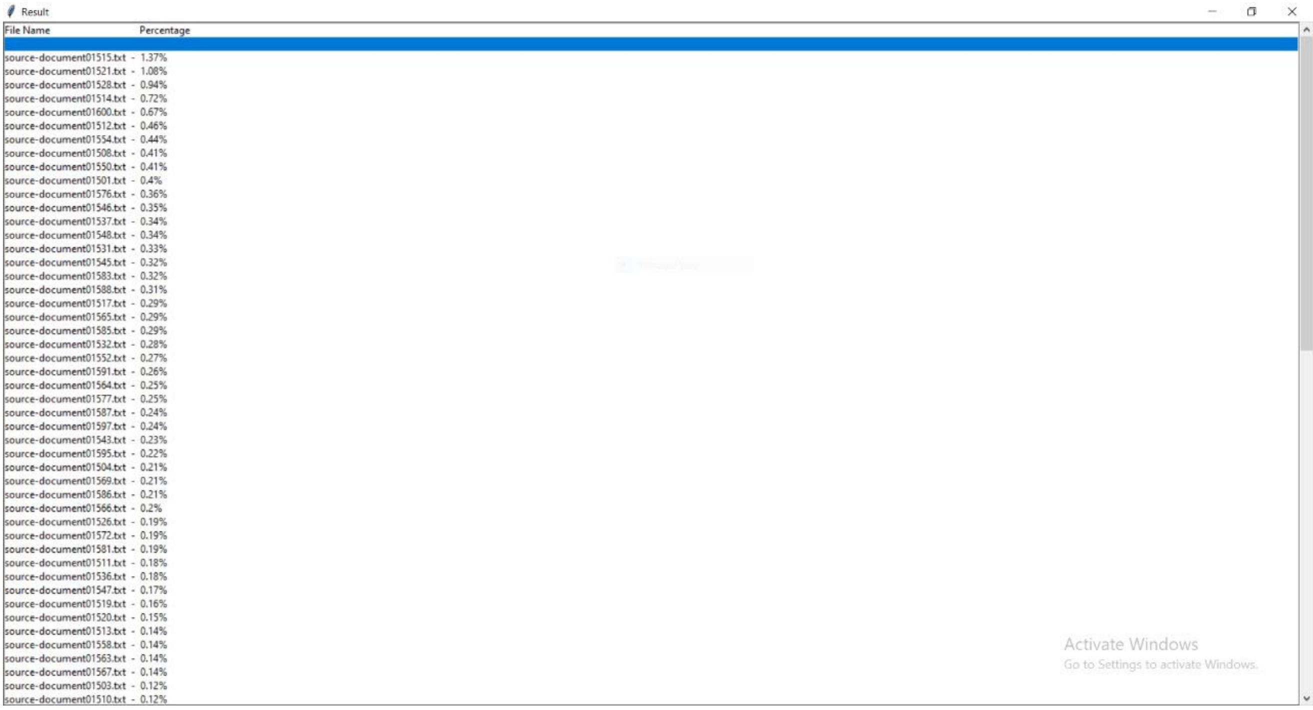
Chapter 6

Project Design





Result:


Result

File Name	Percentage
source-document01515.txt -	1.37%
source-document01521.txt -	1.08%
source-document01528.txt -	0.94%
source-document01514.txt -	0.72%
source-document01600.txt -	0.67%
source-document01512.txt -	0.46%
source-document01554.txt -	0.44%
source-document01508.txt -	0.41%
source-document01550.txt -	0.41%
source-document01501.txt -	0.4%
source-document01576.txt -	0.36%
source-document01546.txt -	0.35%
source-document01537.txt -	0.34%
source-document01548.txt -	0.34%
source-document01531.txt -	0.33%
source-document01545.txt -	0.32%
source-document01583.txt -	0.32%
source-document01588.txt -	0.31%
source-document01517.txt -	0.29%
source-document01565.txt -	0.29%
source-document01585.txt -	0.29%
source-document01532.txt -	0.28%
source-document01552.txt -	0.27%
source-document01591.txt -	0.26%
source-document01564.txt -	0.25%
source-document01577.txt -	0.25%
source-document01587.txt -	0.24%
source-document01597.txt -	0.24%
source-document01543.txt -	0.23%
source-document01595.txt -	0.22%
source-document01504.txt -	0.21%
source-document01569.txt -	0.21%
source-document01586.txt -	0.21%
source-document01566.txt -	0.2%
source-document01526.txt -	0.19%
source-document01572.txt -	0.19%
source-document01581.txt -	0.19%
source-document01511.txt -	0.18%
source-document01536.txt -	0.18%
source-document01547.txt -	0.17%
source-document01519.txt -	0.16%
source-document01520.txt -	0.15%
source-document01513.txt -	0.14%
source-document01558.txt -	0.14%
source-document01563.txt -	0.14%
source-document01567.txt -	0.14%
source-document01503.txt -	0.12%
source-document01510.txt -	0.12%

Activate Windows
Go to Settings to activate Windows.

Chapter 7

Project Scheduling

Group Member	Time	Work to be done
AbhayPratap Singh	2nd week of February	Literature Review, reading papers, Designing UI and choosing algorithm.
Bharat Singh	4rd week of February	Creating the Front end with the text box area and the required buttons.
Aditya Joshi	1st week of March	Learning and testing the functionalities and properties.
Everyone	3rd week of march	Creating a file uploading option with python.
Everyone	4th week of March	Connecting the improved UI with python engine.
Everyone	2nd week of April	Testing and fixing errors and making required changes. Making of Report and Presentation.

Chapter 8

Conclusion

Plagiarism of text has become a common occurrence today with difficult to detect forms such as paraphrasing and summarizing being frequently practiced. Hence, there is a need to design effective mechanisms for automatic plagiarism detection. In this work, a paraphrase recognition approach has been used to detect the occurrence of plagiarism in source and suspicious passages. The system has been tested on three different corpora: Webis CPC, subset of METER, and Wikipedia Rewrite Corpus, at both the passage level as well as the sentence level, with the passage-level approach demonstrating better performance. The system has also exhibited comparable or better performance when compared to the best performing system on the three corpora. Further, the system was also tested on various subcategories of the P4P corpus with good results. This shows that employing paraphrase recognition techniques is a promising direction to explore in the development of plagiarism detection systems.

References

- Hayden Cheers, Yuqing Lin and Shamus P. Smith Academic Source Code Plagiarism Detection by Measuring Program Behavioral Similarity
- John L. Donaldson, Ann-Marie Lancaster and Paula H. Sposato Plagiarism Detection System
- https://www.tutorialspoint.com/python/python_gui_programming.html
- https://github.com/search?q=plagiarism+detector
- https://www.educative.io/edpresso/what-is-sequencematcher-in-python