[Plagiarism Detector]

PROJECT Report

***Submitted by***

# A.Venkata DineshReddy(RA1911028010098

# CH.VamshiKrishna(RA1911028010098

# K.Spurgeon(RA1911028010109

***in partial fulfillment for the award of the degree of***

# B.TECH

***in***

COMPUTER SCIENCE and ENGINEERING

IN

CLOUD COMPUTING

# SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR

[April, 2022]

Introduction

* Plagiarism detection is the process of locating instances of plagiarism or copyright infringement within a work or document.
* The widespread use of computers and the advent of the Internet have made it easier to plagiarize the work of others.
* plagiarism checker can detect plagiarism from billions of web pages.
* This plagiarism check will tell you whether or not your text contains duplicate content
* A plagiarism checker uses advanced database software to scan for matches between your text and existing texts

Libraries Used

* Flask
* Pandas
* Difflib
* Natural language toolkit (NLTK)
* Beautiful Soup
* Request

Working Process

The UI features a simple text area input in which the user can paste the text from an article. When the user clicks the Submit button, this input is used to query a database of articles. Results and their match scores are then displayed to the user. To help reduce the amount of noise, the app also includes a slider input in which the user can specify a similarity threshold to only show extremely strong matches.

As you can see, when original content is used as the search input, the match scores for possibly plagiarized articles are relatively low. However, if we were to copy and paste the text from one of the articles in our database, the results for the plagiarized article come back with a 99.99% match!

This is done using Python Flask that utilizes the Pinecone SDK. The HTML uses a template file, and the rest of the frontend is built using static CSS and JS assets.

Advantages

* The system designed to detect similarity among text documents calculates content similarity among specified documents.
* Applying efficient string-matching algorithm, which will further reduce the time and increase the efficiency.
* This system is simple and easy to access.
* System will give accurate results based on the content provided.
* This system will generate results in very less time.
* Code is short and simple to understand

Disadvantages

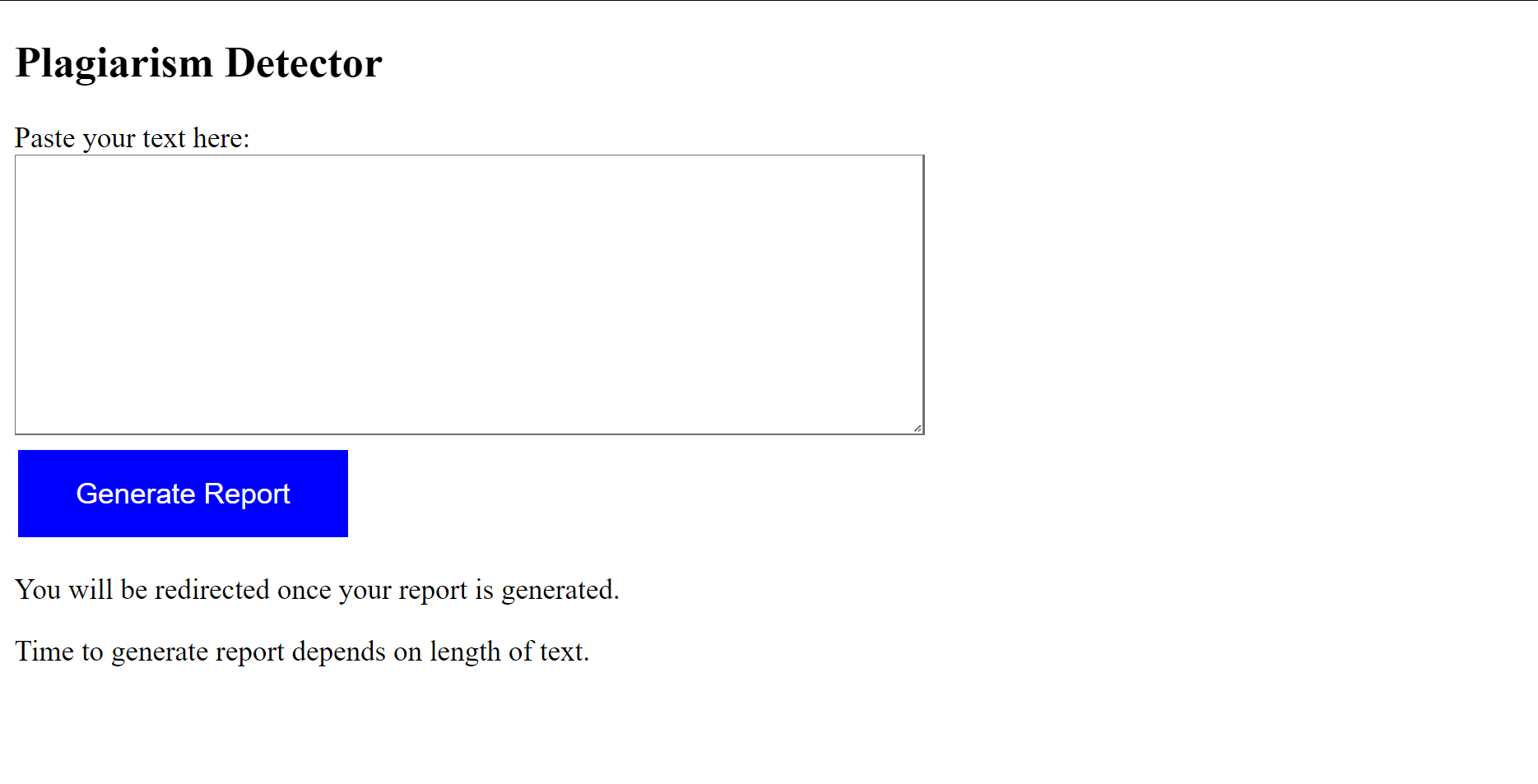
* Active internet connection required.
* If user uploads an incorrect document, then the result won’t be accurate.
* Takes more time to get results if input data is large.
* Input should give as text and docx, pdf formats are not supported.
* The plagiarism tool can accurately detect plagiarism when the text is comprised of seven or more than seven words but it cannot work as well with the text of smaller words.

Deployment of Advanced Algorithms

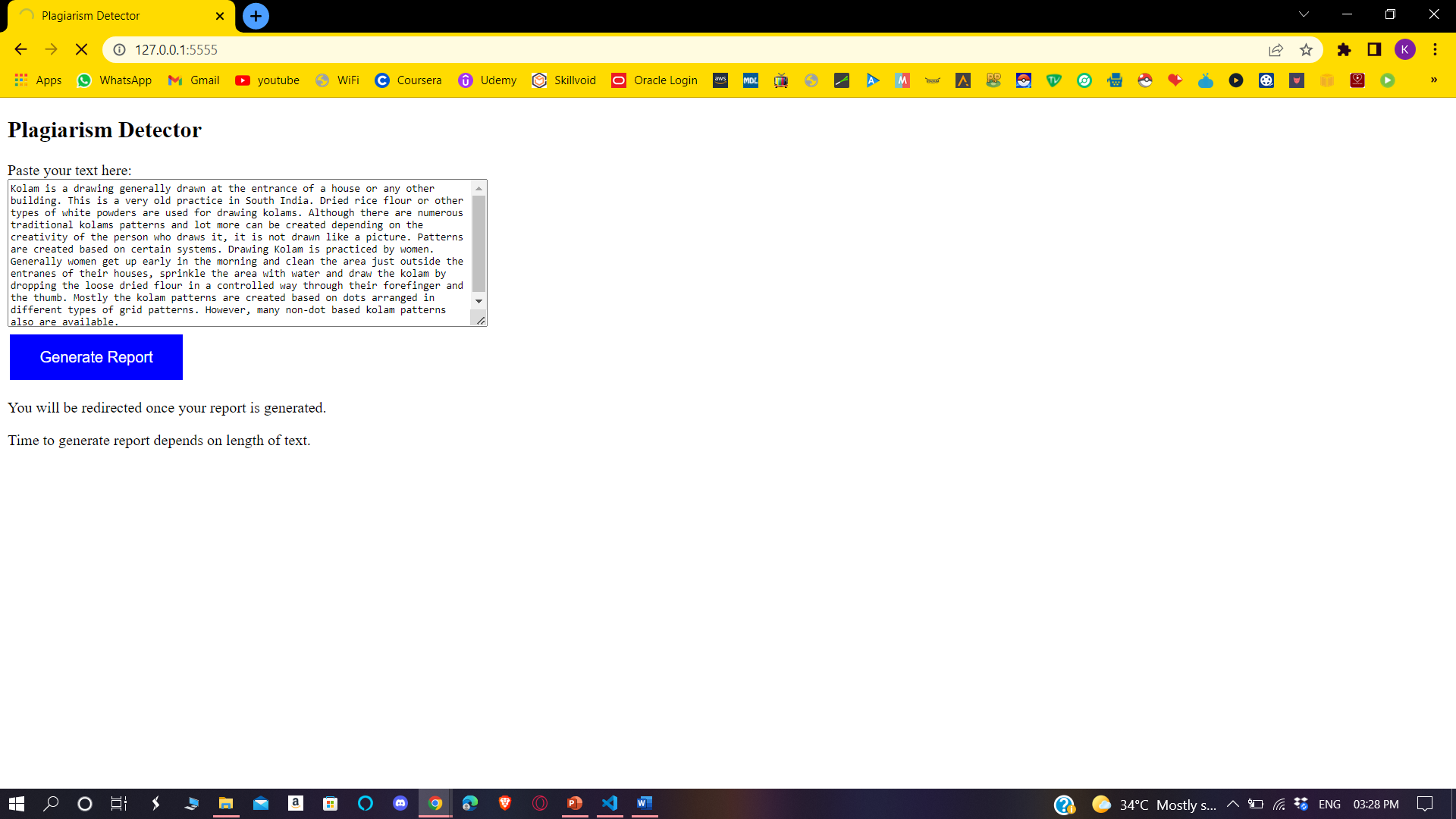
1. Algorithms are the identity of AI. For comparing two texts the system contributes some advanced algorithms. These algorithms take the sample content and analyze the words in sequence.
2. Then word by word AI algorithms match the entire text with the millions of webpages on the internet. If the content is found to be similar or the same, the detector shows the error.
3. Karp-Rabin algorithm, string matching algorithm are used in our project.

Outputs

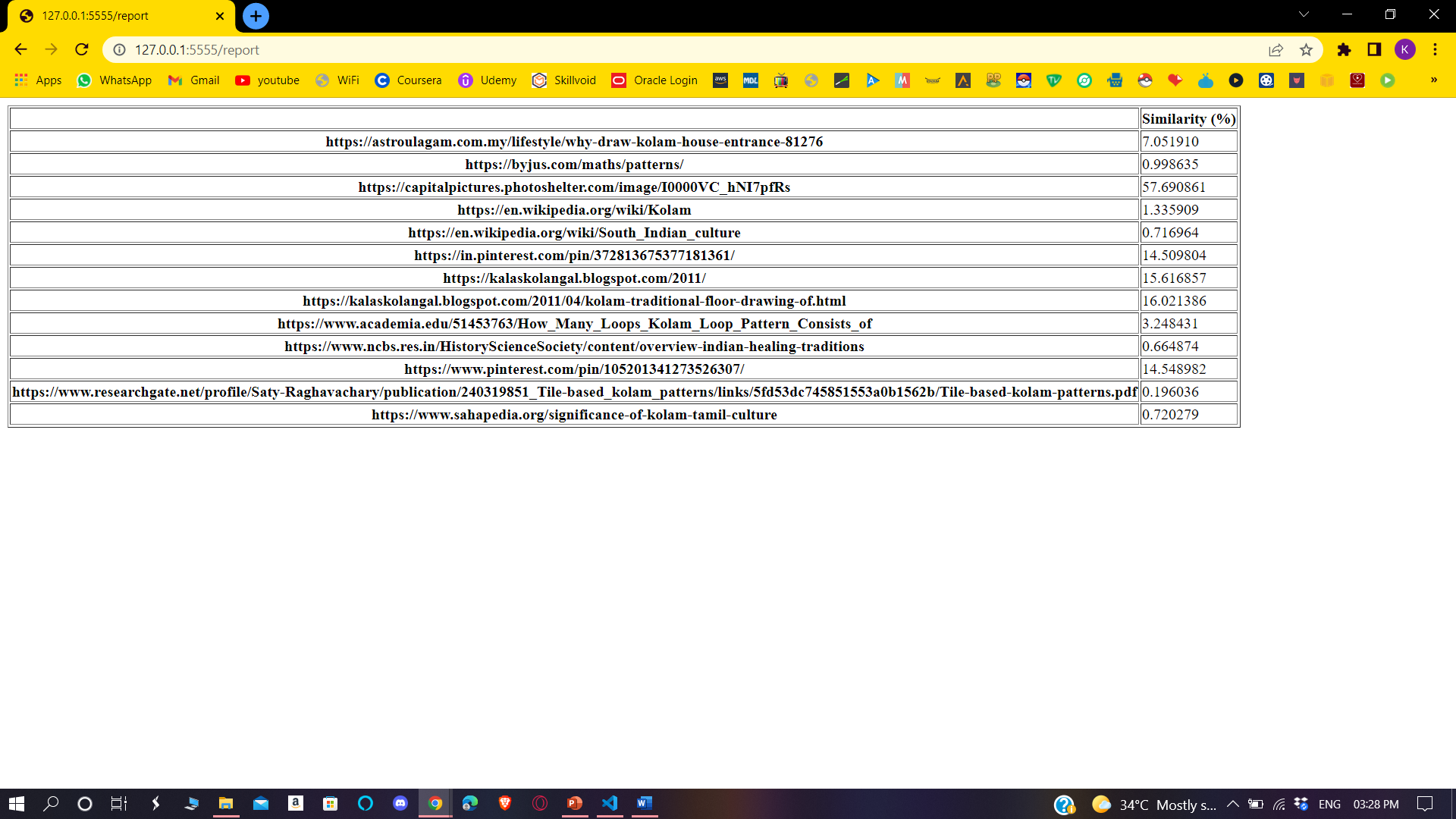
When we run all the files of code and the main file, then we will get the html page as like this



=>Let’s take an example and copy some text from a site and check the plagiarism of that text



This shows the percentage from which site, the text has been copied.



Conclusion

1. Plagiarism is rampant now. With most of the data available to us in digital format the venues for plagiarism is opening up.
2. To avoid this kind of cheating and to acknowledge the originality of the author, new detection techniques are to be created.
3. To protect the intellectual property source code new techniques are to be developed and implemented.
4. So, this is one of the technique which is used to find out the source site from which you have copied the matter.