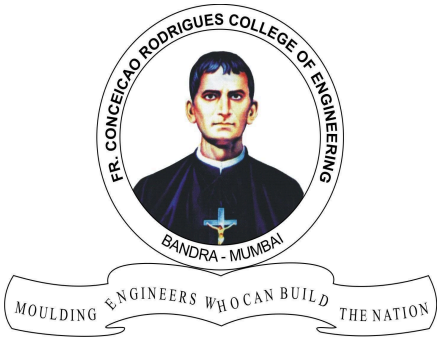
Plag Detection

A project report submitted in partial
fulfillment of the requirements for the
degree of

Bachelor of Engineering
In
Electronics and Computer Science

by
Shubham Soni Nath (9143)
Alisha Rawat (9154)
Leory Machado (9138)

Under the guidance of
Prof. Prajakta Bhangale
(Assistant Professor)



DEPARTMENT OF ELECTRONICS AND COMPUTER SCIENCE
Fr. Conceicao Rodrigues College of Engineering, Bandra (W), Mumbai - 400050
University of Mumbai
(2023-24)

This work is dedicated to my family.

I am very thankful for their motivation and support.

Internal Approval Sheet

CERTIFICATE

This is to certify that the project entitled **"ProjectHub - Centralized Project Repository"** is a bonafide work of **Shubham Soni Nath (9143) , Alisha Rawat (9154), Leroy Machado (9138)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of Bachelor in **Electronics and Computer Science.**

(Prof. Prajakta Bhangale)
Supervisor/Guide

(Dr. Sapna Prabhu)
Head of Department

(Dr. S. S. Rathod)
Principal

Approval Sheet

Project Report Approval

This project report entitled **ProjectHub - Centralized Project Repository** by **Shubham Soni Nath, Alisha Rawat, Leroy Machado** is approved for the degree of Bachelor of Engineering in Electronics and Computer Science.

Examiner 1. ———————————

Examiner 2. ———————————

Date:

Place:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Shubham Soni Nath (9143) **(sign)** _____

Alisha Rawat (9154) **(sign)** _____

Leroy Machado (9138) **(sign)** _____

Date:

Abstract

ProjectHub is a centralized project repository designed to manage the academic projects and research of the students and to encourage collaborative work. Along with the repository, the project particularly focuses on the plagiarism detection system, which employs a combination of cosine similarity and jaccard similarity. In modern educational institutes, it is very important to maintain academic integrity among the students when they undertake different projects. In this project, two very fundamental text analysis techniques are being used which play a very pivotal role in determining whether a certain document is plagiarized or not. Cosine similarity keeps in check with the semantic plagiarism while Jaccard similarity deals with the structural plagiarism of the documents. Thus, by combining the results of these two algorithms, we can enhance the project's overall accuracy and credibility. This report also provides insights into the implementation of the repository and the user interface thereby, ensuring the quality, integrity and authenticity of academic work.

Acknowledgments

We have great pleasure in presenting the report on "ProjectHub - Centralized Project Repository". We take this opportunity to express our sincere thanks towards the guide Prof. Prajakta Bhangale, Fr. C.R.C.E, Bandra (W), Mumbai, for providing the technical guidelines, and the suggestions regarding the line of this work. We enjoyed discussing the work progress with her at every step of our project.

We thank Dr. Sapna Prabhu, Head of Electronics and Computer Science Engineering Dept., Principal and the management of C.R.C.E., Mumbai for encouragement and providing necessary infrastructure for pursuing the project.

We also thank all non-teaching staff for their valuable support, to complete our project.

Shubham Soni Nath(Roll No.9143)

Leroy Machado (Roll No. 9138)

Alisha Rawat (Roll No.9154)

Date:

Contents

List Of Figures

List Of Tables

Chapter 1
Introduction

A brief understanding of the topic and the motivation behind why the topic was chosen is presented in the following chapter. The chapter also specifies the various problems that may have arisen without this project and how this project has made the users (students and teacher) life much more easy and comfortable.

1.1 Introduction

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Amet consectetur adipiscing elit ut. Tempus imperdiet nulla malesuada pellentesque. Turpis cursus in hac habitasse. Tortor at auctor urna nunc id. Nibh praesent tristique magna sit. Ipsum consequat nisl vel pretium lectus quam id. Dis parturient montes nascetur ridiculus mus mauris vitae. Mauris rhoncus aenean vel elit. Dictumst quisque sagittis purus sit amet volutpat. Porta lorem mollis aliquam ut porttitor leo. Gravida cum sociis natoque penatibus et. Duis at tellus at urna condimentum. Est ullamcorper eget nulla facilisi etiam dignissim diam quis enim. Luctus venenatis lectus magna fringilla urna porttitor.

Mauris commodo quis imperdiet massa tincidunt. Mi ipsum faucibus vitae aliquet nec ullamcorper sit amet. Magna eget est lorem ipsum dolor sit amet consectetur adipiscing. Eget duis at tellus at urna. Amet nisl suscipit adipiscing bibendum. Id nibh tortor id aliquet lectus proin nibh nisl condimentum. Est sit amet facilisis magna etiam tempor orci eu. Pharetra massa massa ultricies mi quis hendrerit dolor magna. Amet massa vitae tortor condimentum lacinia quis vel. Nibh mauris cursus mattis molestie a. Sed elementum tempus egestas sed sed risus pretium quam. Faucibus in ornare quam viverra orci sagittis eu.

Proin nibh nisl condimentum id venenatis a condimentum. Vitae tempus quam pellentesque nec nam aliquam sem et tortor. Ut tortor pretium viverra suspendisse potenti. Ut placerat orci nulla pellentesque dignissim. Velit scelerisque in dictum non consectetur a erat. Lobortis scelerisque fermentum dui faucibus in ornare. Euismod quis viverra nibh cras. In metus vulputate eu scelerisque felis. Ultricies leo integer malesuada nunc vel risus commodo viverra. Ridiculus mus mauris vitae ultricies leo integer malesuada. Purus faucibus ornare suspendisse sed nisi lacus. Gravida arcu ac tortor dignissim. Id consectetur purus ut faucibus pulvinar elementum integer enim neque.

Id aliquet lectus proin nibh nisl condimentum id venenatis a. Nisi est sit amet facilisis magna etiam tempor. Amet tellus cras adipiscing enim eu turpis. Sapien pellentesque habitant morbi tristique. Id donec ultrices tincidunt arcu non sodales neque. Tempor id eu nisl nunc mi ipsum faucibus. Faucibus vitae aliquet nec ullamcorper sit amet risus nullam. Dolor morbi non arcu risus. Congue eu consequat ac felis donec et odio. Iaculis at erat pellentesque adipiscing commodo elit at imperdiet. A pellentesque sit amet porttitor eget dolor morbi. Pellentesque nec nam aliquam sem et. Suscipit adipiscing bibendum est ultricies integer. Sem nulla pharetra diam sit amet nisl suscipit. Enim diam vulputate ut pharetra sit. Facilisi morbi tempus iaculis urna id volutpat. Posuere sollicitudin aliquam ultrices sagittis orci a scelerisque.

1.2 Motivation

The motivation behind developing the Centralized Project Repository for engineering college students with an integrated plagiarism checker stems from a deep understanding of the

challenges faced by both students and educators in the realm of academic research and project management. In the contemporary educational landscape, the volume of academic content being generated is vast, making it increasingly difficult to maintain the quality, originality, and integrity of student projects[1]. By incorporating a robust plagiarism checker, students are encouraged to develop a strong sense of ethics, ensuring that their work is a true reflection of their knowledge and skills.Also having access to a diverse range of projects, students can learn from their peers, gain inspiration, and collaborate on innovative ventures. This collaborative environment fosters a culture of creativity and mutual learning, enhancing the overall academic experience. The digital era demands technological proficiency and ethical awareness. Our motivation is to equip students with the skills and values that define success in the modern world. Through the integration of a plagiarism checker, we not only emphasize the importance of originality but also introduce students to cutting-edge technologies[3]. This exposure prepares them for a future where technological fluency and ethical conduct are indispensable.

In essence, our motivation for the Centralized Project Repository for engineering college students with a plagiarism checker goes beyond mere software development. It is a commitment to nurturing a generation of thinkers, innovators, and ethical leaders. It is a dedication to creating an educational ecosystem where every student's journey is marked by integrity, collaboration, innovation, and preparedness for the challenges that lie ahead.

1.3 Problem Statement

The lack of a centralized platform to consolidate and share information about projects undertaken by students in technical colleges, hinders the potential for peer learning , cross-functional research and innovation. Therefore, there is a need to develop an integrated online platform that facilitates knowledge sharing, promotes innovation, and ensures academic integrity by offering plagiarism detection capabilities.

1.4 Objectives

The objectives of our project are :

1. To serve as a common knowledge repository, enabling peer learning and facilitating cross-functional research between different departments in the college.
2. To identify and explore the best algorithm which will be suitable for our project.
3. To develop an application that uses two algorithms parallelly for plagiarism detection, thereby optimizing the overall efficiency and effectiveness.

Chapter 2
Literature Review

To gain a much deeper understanding of our project and to decide on which algorithms will work efficiently together and give more accurate results, we have surveyed and read many papers of topics ranging from ‘what are the different plagiarism detection types’[1][2] to ‘which plagiarism algorithm works well under this specific type’[3][4][5][6][7][8][9][10][11][12] to ‘the different methodologies we can implement in our repository’[13][14][15][16].

- Comparison of Plagiarism Detection Methods

[1]The paper states the different types of plagiarism, it describes the various forms of plagiarism detection and the different plagiarism detection methods available. The paper also gives information about the various detection tools present and states their comparison. The paper also presents details and importance of syntactic and semantic level plagiarism detection for both text document and source code programs.

[2]The paper explores three approaches for plagiarism detection: Vector representation, which transforms text into vectors while preserving semantic and syntactic aspects; Level treatment, which determines the depth of text treatment; and Similarity methods, which maintain the semantic aspect of the text. However, these methods often conduct similarity analyses on a word-by-word or sentence-by-sentence basis, potentially leading to unreliable results. This is because two documents may share identical words or sentences without being semantically similar. Additionally, the semantic aspect can be lost when documents are processed as lists of sentences or words.

Paper [3] compares the syntactic search engine and the semantic search engine. The Syntactic Web relies on keyword matching for results, while the Semantic Web enhances search by comprehending queries, resulting in more meaningful and specific outputs. Experimental evidence indicates that the semantic search engine outperforms syntactic search engines in retrieving relevant documents, as measured by precision and recall metrics.

[4]The paper focuses on plagiarism detection using semantic analysis. Broadly, the proposed system comprises three primary stages, each encompassing multiple steps. The initial stage involves the pre-processing stage, which consists of tokenization and stop words removable. Second stage is the document specialization stage. The concluding phase involves semantic plagiarism detection, where the degree of semantic plagiarism is assessed. WordNet is utilized to identify synonyms for each word, treating these synonyms as equivalent to the original word when identifying instances of plagiarism.

Paper [3] gives us a descriptive comparison of syntactic and semantic plagiarism detection. Basically, syntactic plagiarism just focuses on the syntax of the sentence whereas semantic plagiarism detection relies more on the logic of the sentence than just the syntax of the sentence. So, even if the order of the words or the structure of the sentences are changed, this detection method will identify it as plagiarized. And, that is why we will be using an algorithm that helps in semantic plagiarism[4].

- Plagiarism Detection Algorithm: Cosine Similarity

[5][6] This paper's proposed system employs string searching algorithms to identify matching content in plagiarized documents, with the Rabin-Karp Algorithm being the chosen method. The algorithm relies on hash values to quickly compare patterns with substrings in the text. Hash collisions, where two different substrings produce the same hash value, can lead to false positives. To mitigate this, additional checks are needed to confirm matches, which can slow down the algorithm. Other disadvantages of using this algorithm are that this is not designed for semantic analysis and it is sensitive to minor text changes.

[7] The paper commences by employing the term frequency-inverse document frequency (TF-IDF) algorithm as the vectorization method. Preprocessing of the data precedes its input into the model, wherein sentence tokenization and stop word elimination are performed. Subsequently, similarities among sentences are computed to detect plagiarism within the document. The study further evaluates several similarity measurement algorithms, including cosine similarity, Levenshtein edit distance, Manhattan distance, and Jaccard similarity.

Through this examination, it is deduced that the cosine similarity algorithm outperforms others in terms of speed and accuracy when analyzing Bengali sentences containing multi-byte characters. Additionally, it is noted that cosine similarity effectively identifies semantic parallels and captures paraphrasing and rephrasing.

[8] This paper introduces a plagiarism detection system employing machine learning components such as word2vec and cosine similarity. The system operates effectively, accurately identifying the level of plagiarism across provided text files. Additionally, the inclusion of a user interface enhances accessibility, making the system more user-friendly for non-experts.

[9] This paper specifies the steps used in the preprocessing stage such as tokenization, stop-word removal and stemming. For string matching algorithm, Knuth-Morris-Pratt Algorithm is used. It checks the characters from left to right. However, KMP requires additional space for the partial match table. In cases where memory usage is a critical concern, this extra space may be a limitation. Other limitations include that this algorithm is not designed for semantic analysis and is limited in capturing paraphrasing.

Algorithm	Advantages	Disadvantages
Rabin-Karp[5][6]	1. Efficient for exact matching. 2. Fast for short patterns in long documents.	1. Not designed for semantic analysis. 2. Sensitive to minor text changes.
Cosine Similarity[7][8]	1. Identifies semantic similarities. 2. Captures paraphrasing and rephrasing.	1. May not work well for very short documents.
Knuth-Morris-Pratt [9]	1. Efficient for exact string matching. 2. Handles large documents with minimal memory.	1. Not designed for semantic analysis. 2. Limited in capturing paraphrasing.

Table 2.1 Comparison for semantic plagiarism algorithms

Thus, from our project point of view Cosine Similarity[7][8] matches our preferences and has several advantages over Robin-Karp[5][6] and Knuth-Morris-Pratt[9] algorithms. And that is why, we will be using the cosine similarity algorithm.

In order to increase accuracy and efficiency, we plan to carry out plagiarism detection from two different algorithms parallelly. One that performs semantic plagiarism detection and the other that performs structural plagiarism. Now why structural plagiarism, because this method detects synonyms of words between the documents, if the writer may have substituted synonyms for the words and kept the structure of the sentence the same[10].

- Plagiarism Detection Algorithm: Jaccard Similarity

[11] The primary contribution of this paper lies in the introduction of a novel plagiarism detection system capable of addressing various forms of lexical, syntactic, and semantic plagiarism. It involves the extraction of negative non-plagiarized and positive plagiarized instances from benchmark dataset documents containing diverse plagiarism types. The chi-square algorithm is employed to rank the thirty-four features within the training database, extracting the most discriminative ones that yield the highest detection accuracy. Additionally, the SVM classification algorithm is utilized to establish the hyperplane equation for selected features, offering the ability to introduce new dimensions to differentiate between overlapping training cases of different classes. However, it is noted that SVM necessitates labeled training data and can be computationally demanding for large datasets.

[12] The paper explores the creation of a plagiarism detection tool for Tetun language employing a text mining methodology. This tool employs the Jaccard similarity coefficient to quantify the degree of similarity among processed thesis titles. The experiment results show that the system performs well in detecting plagiarism, with a precision of 0.90 and a recall of 0.94. The Jaccard similarity coefficient is used in comparing two samples or documents to determine their similarity. Overall, the research aims to address the need for a plagiarism detection system that can support the Tetun language in academic settings. Advantages of using Jaccard Similarity are that it is robust to document length variation, it has fast computation and is well-suited for identifying common phrases.

Algorithm	Advantages	Disadvantages
SVM[8]	1. Effective for high-dimensional data. 2. Can capture complex patterns and relationships in data.	1. Requires labeled training data. 2. Computationally expensive for large datasets.
Latent Semantic Analysis (LSA/LSI)[8]	1. Captures semantic relationships. 2. Can identify related terms and concepts.	1. Requires SVD decomposition, which can be resource-intensive. 2. Requires a large corpus for training.
Jaccard Similarity[12]	1. Robust to document length variations 2. Fast computation. 3. Well-suited for identifying common phrases.	1. Ignores Term Frequency. 2. Limited to Set-based Comparison.

Table 2.2 Comparison for structural plagiarism algorithms

Thus, from our project point-of- view Jaccard Similarity[12] matches our preferences and has several advantages over SVM[11]. And that is why, we will be using the jaccard similarity algorithm.

- Project Repositories and Web Search Engines

Now, apart from plagiarism detection, our project also focuses on project repositories and web search engines. These will be further discussed below.

[13] The project follows a Full Stack web development approach, utilizing JavaScript, HTML, and CSS for the front-end, Node.js for the back-end, and MongoDB for the database. It incorporates features such as displaying past student projects, showcasing personal research work, enabling collaboration, and offering additional learning resources.

[14] The methodology involves creating and managing a user-friendly database for faculty remarks, automating student feedback, providing easy document access, enhancing usability with JavaScript, selecting between MySQL or Firebase for project data storage, ensuring data backup via Google Cloud Platform, and ultimately improving project management efficiency in university settings. Challenges include data security. Also, internet connection to ensure backup

on Google Cloud Platform.

[15] The methodology employed in this paper entailed the creation of a web-based digital project repository accessible to all researchers in the Sultanate. It utilized Bootstrap and the MVC-based PHP Laravel Framework to create an easily navigable platform tailored for student use and educational institutions to manage, categorize, and exhibit research projects, fostering open sharing and showcasing of research work.

[16] This paper conducts an analysis of overseas studies regarding the deposition, management, and utilization of publicly funded research papers. Furthermore, it introduces a project proposal aimed at establishing a national repository with open access principles to facilitate the integration and centralized management of such research papers at a national level. To establish and effectively employ a nationwide repository grounded in open access, it is imperative to undertake thorough actions spanning from enhancing legal frameworks and policies to implementing utilization systems, including: 1. Crafting Laws and Regulations, Public Deposit Policies; 2. Standardization, both Nationally and Globally Networking; 3. Developing a Research Knowledge System based on Repositories; 4. Implementing Technology for Deep Learning and Text Mining of Data.

Chapter 3
Proposed System

3.1 Introduction

ProjectHub is a centralized repository that is developed to bring together a diverse range of projects undertaken by students of different branches in a college. This centralization of knowledge serves as a valuable resource for students, educators, and researchers alike. This will create a comprehensive collection of projects from various sources. The platform is incorporated with plagiarism detection tools that ensures the authenticity and originality of the projects that students will be working on. This feature will help to maintain the integrity of the platform and will also encourage students to develop unique and innovative projects.

ProjectHub is to be accessed by students with credentials that they use to their college portal system. After successfully login they can search for previously made projects and have an understanding of what kinds of projects are to be made and how previously students saw a problem and gave solutions. They choose a project from that with an intention to make it in a more novel way or give a different approach that would provide even better results. The other way is to make a new fresh start where they propose their own idea and the system by doing plagiarism checks informs them whether they can proceed with that idea or not because that idea has already been overtaken by previous students. This will help students to be more creative and innovative with their thinking process. ProjectHub solves the problem of redundant projects and also won't let students use projects that were already made by previous year college students. If students try to use exactly previously made projects with any attempt of innovation then the plagiarism checks on ProjectHub won't let students proceed. This upholds the principles of academic integrity, the platform incorporates robust plagiarism detection tools. These tools help ensure that all projects proposed are original and have not been copied from elsewhere. The frontend of the system is based on React.js which is a well known and most widely used framework of JavaScript. React.js is a JavaScript library that is used here to develop the user interface through which all the user interaction will be handled and all the requests to the server

will be sent to send input data which will be in text as well as other multimedia formats. The proposed system is dynamic and responsive to all screen sizes making it available to all platforms that supports any kind of latest web browser. To create a responsive and user-friendly UI, Tailwind CSS is used for efficient development and clean code.

The backend of the system is structured around Flask, a lightweight and flexible Python web framework. Flask handles server-side JavaScript execution and facilitates API development for effective communication with the frontend. It is employed for routing and middleware creation, streamlining CRUD operations.

The plagiarism detection functionality is implemented using Python, capitalizing on its diverse libraries to expedite development and enhance overall functionality. The Python-based cosine and Jaccard similarity models are deployed on a Flask server, establishing seamless communication within the system. This approach prioritizes simplicity and customization, aligning with Flask's design principles.

3.2 Algorithm and Process Design Flow

User flow following steps to do plagiarism check for their project:

Step 1: Authentication

Students login with their username and password provided by the college.

Step 2: Upload Proposal or Report

Students will upload the proposal/report in PDF format or in text format.

Step 3: Plagiarism check

Plagiarism check will follow below procedure involving plagiarism detection using both cosine and jaccard similarity algorithms parallelly with the help of ensemble learning.

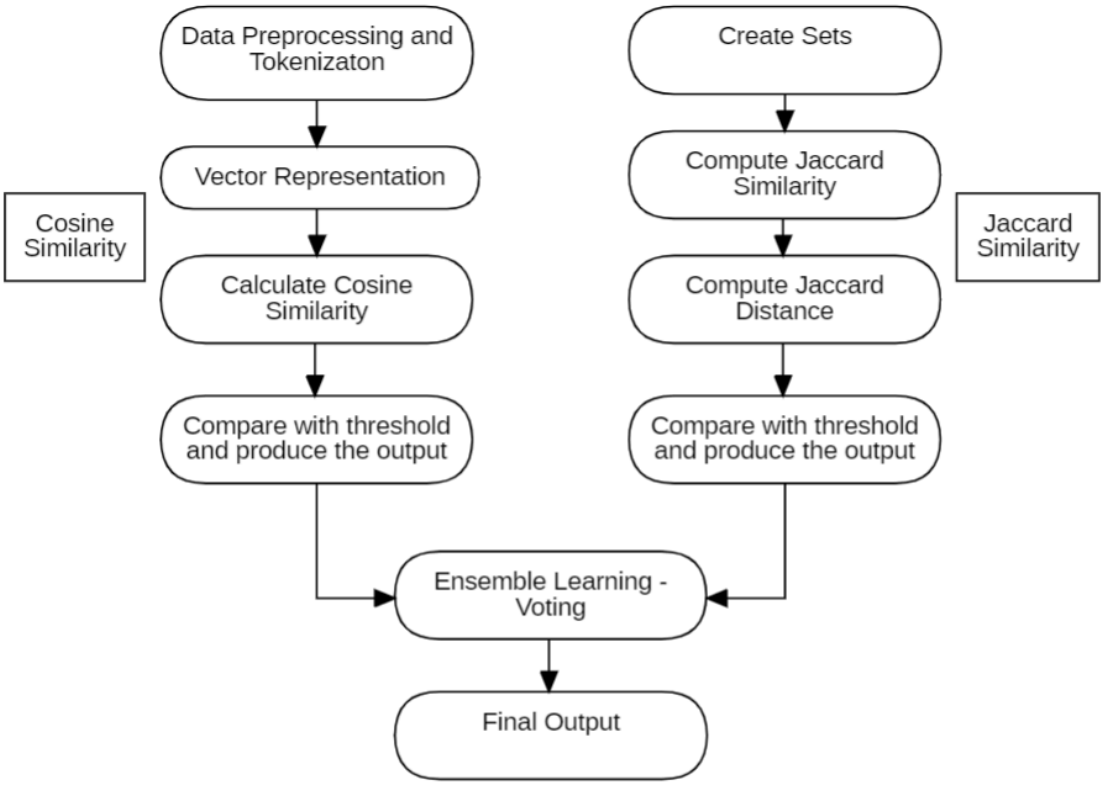


Figure 3.2.1 Process Design Flow

For Cosine Similarity:

Step 1a : Data Preprocessing and Tokenization

This step involves the processing of the data before computing the cosine value, this includes cleaning and preparing the data, by removing punctuations and stop words. This step also involves converting text data into tokens.

Step 2a : Vector Representation

Since Cosine Similarity deals with vectors, we need to convert the tokens (from step 1) into vectors by using the TF-IDF technique, which will further help in the mathematical calculations.

Step 3a : Calculate Cosine Similarity

Cosine Similarity is used to find the angle between two vectors. Larger the angle between the two vectors, lesser is the similarity. These angles range from 0o to 180o.

The formula for cosine similarity is,

$$D(x, y) = cos(\theta) = \frac{x.y}{||x||\,||y||}$$

where,

- x.y is the dot product of the two vectors ‘x’ and ‘y’.
- x and y are the magnitude of the two vectors ‘x’ and ‘y’.

The value of Cosine Similarity ranges from 0 to 1 , where 0 indicates that the vectors, in other words, the documents are similar, while 1 means there is nothing common between the vectors, i.e. the documents.

In this step, we calculate the cosine similarity with the help of the formula using the vectors generated in step 2. The output of this step will be in the range of 0-1.

For Jaccard Similarity:

Step 1b : Create Sets

Since Jaccard Similarity deals with sets, we need to convert the text data into tokens and then convert the tokens into sets of words thus each document becomes a set of words.

Step 2b : Compute Jaccard Similarity

Jaccard Similarity is used to find the distance between two data sets.

The formula for jaccard similarity is,

$$JIndex = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cup B|}$$

Step 3b : Compute Jaccard Distance

This step is very essential to the Jaccard similarity algorithm.

$$JD(A, B) = 1 - JIndex$$

The value of Jaccard Similarity ranges from 0 to 1 , where 0 indicates that the sets, in other words, the documents are similar, while 1 means there is nothing common between the sets, i.e. the documents.

Step 4 : Compare with threshold and produce output

The output generated from both algorithms will be compared with their respective threshold value which will help us detect if the text document is plagiarized or not.

Step 5: Combining Results

The results from the cosine similarity and Jaccard similarity algorithms represent the individual "votes" from each model. By combining these results, we create a more comprehensive understanding of the document similarities. This combination could be achieved through various methods such as averaging, weighted averaging, or any other suitable aggregation technique. The goal is to leverage the strengths of both similarity metrics, considering their different

perspectives on document similarity.

Step 6: Voting Mechanism for Final Decision

Once the results are combined, a voting mechanism is implemented to make the final decision on whether plagiarism is detected or not. The thresholds for both cosine similarity and Jaccard similarity act as the criteria for individual votes. The ensemble then "votes" based on the combined result, determining the final output to be displayed to the user. This voting process ensures that the decision is not overly influenced by one metric and provides a more balanced and robust outcome.

In summary, the ensemble learning approach in this code involves combining the individual votes from cosine and Jaccard similarity algorithms and then employing a voting mechanism to make the ultimate decision on plagiarism detection. This methodology enhances the overall reliability and accuracy of the plagiarism detection system by leveraging the complementary strengths of the two similarity metrics.

3.3 System Requirements

Operating System:	Any OS is compatible
Processor:	Intel i5 (7th gen) or later
Memory(RAM):	8 GB minimum, 16 GB recommended
GPU:	GTX1050 or later
Memory(VRAM):	4GB or more
Internet connection is only required to download libraries	

Table 3.1 System requirements

3.4 Details of Software

1. Operating System: Windows 7 or higher

   For running the neural network model either windows 7 or newer version is required to run.

2. Jupyter Notebook or Google Colab

   Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations, and narrative text. The notebook interface allows you to write and run code in a browser-based environment. Apart from Jupyter Notebook, Google Colab can be used which is a cloud platform that provides environment and necessary hardware.

3. Python 3.x

   Python is a widely used programming language in the field of Artificial Intelligence (AI) due to its simplicity, versatility, and availability of a wide range of libraries and frameworks that make it easy to develop AI applications[16].

4. NLTK (Natural Language Toolkit)

   NLTK is used for text preprocessing tasks in the provided code. It includes tokenization through the word_tokenize function, which breaks down the text into individual words, making it suitable for analysis. Additionally, NLTK's stopwords module helps remove common stop words from the text to reduce noise, allowing a focus on more meaningful words.

5. scikit-learn (sklearn)

   Scikit-learn is utilized for advanced text analysis. It provides the TfidfVectorizer for converting the preprocessed text into TF-IDF vectors, which quantify the importance of words in a document relative to a corpus. Additionally, the cosine_similarity function from scikit-learn calculates the cosine similarity between TF-IDF vectors. This similarity

measure is widely used in text analysis to assess the similarity between documents or text passages. Together, NLTK and scikit-learn offer a comprehensive solution for text preprocessing, vectorization, and similarity calculation, ideal for various text analysis tasks, including plagiarism detection.

6. Flask

   Flask is a lightweight, open-source Python web framework emphasizing simplicity and flexibility. Ideal for small to medium projects, it provides essential tools for web development, leaving room for developer choices. With a minimalistic design, Flask boasts easy URL routing, Jinja2 templating, and an extensible architecture through various extensions. Leveraging the Werkzeug WSGI toolkit, it includes a built-in development server for seamless testing. Considered a "micro" framework, Flask empowers developers to shape their applications with preferred tools, making it a popular choice for those valuing simplicity and customization in web development.

Chapter 4
Results

The documents with different complexities are tested and analyzed through which confusion matrix is made.

Confusion Matrix	True(Cosine)	False(Cosine)
True(Jaccard)	Gives output as Plagiarized	If the Cosine similarity value is less than cosine threshold then not plagiarized or if the value is greater, then structural plagiarism exists so plagiarized.
False(Jaccard)	If the Jaccard similarity value is less than jaccard threshold then not plagiarized or if the value is greater then semantic plagiarism exists so plagiarized.	Gives output as not Plagiarized

Each document was subjected to both algorithms, and the resulting similarity scores were recorded. We then manually labeled a subset of documents as plagiarized or non-plagiarized for comparison purposes.

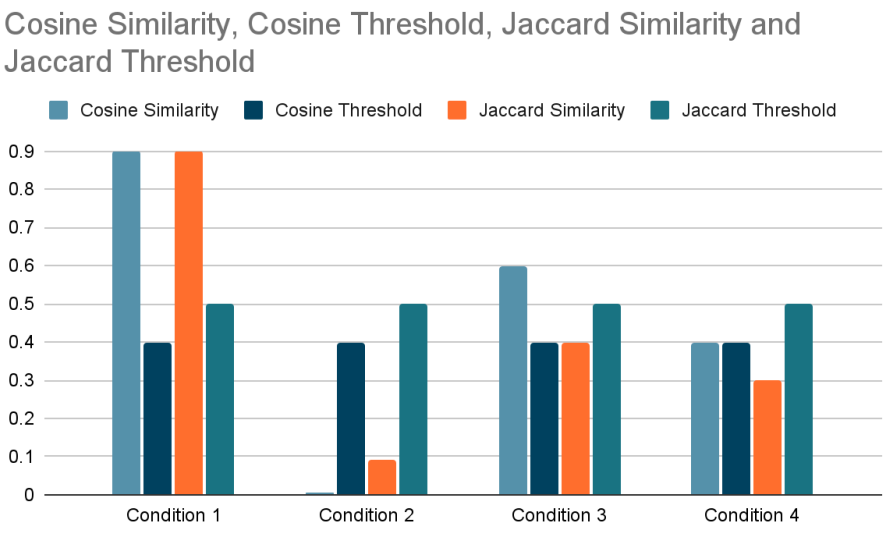


Figure 3.2.1 Process Design Flow

Paper	Cosine Similarity	Jaccard Similarity
Aparna Babu, Anju S, Archana P Udayan, Giny Mary John and Divya S B, “Online Plagiarism Detection and Result Evaluation using Data Mining and NLP”, International Journal of Engineering Research & Technology (IJERT) 2022.	✔	×
El-Rashidy, M.A., Mohamed, R.G., El-Fishawy, N.A. et al. An effective text plagiarism detection system based on feature selection and SVM techniques. Multimed Tools Appl 83, 2609–2646 (2024). https://doi.org/10.1007/s11042-023-15703-4	×	✔
ProjectHub	✔	✔

4.1 Display Of Result

Following figures are the end result. Here we have successfully implemented our system.

Here the detector is based on a cosine similarity algorithm which is detecting semantic similarity for the input data with the data present in the database.

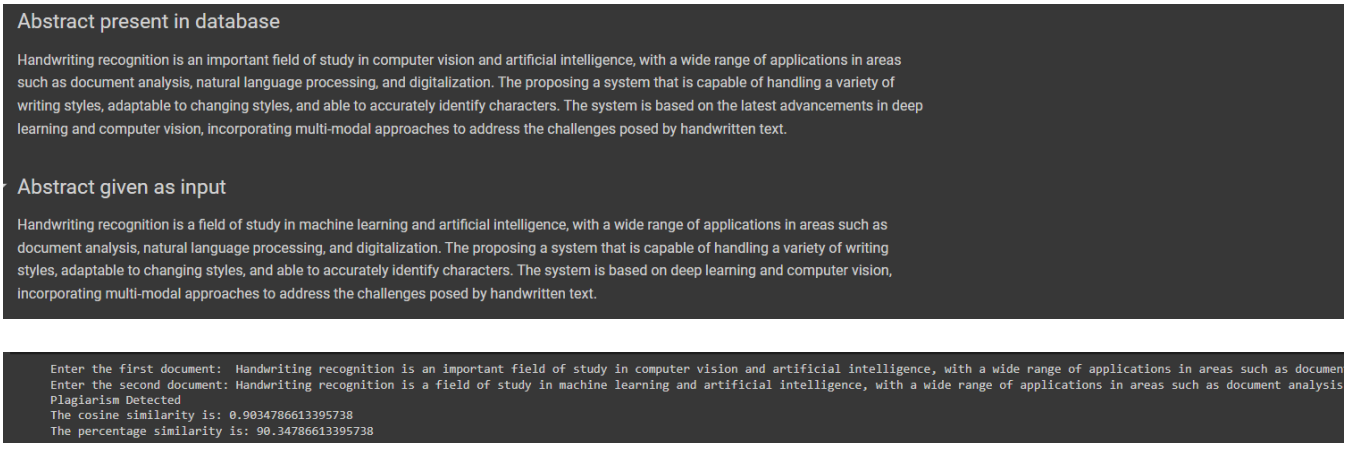


Figure 4.1.1 Result of Cosine Similarity 1

In fig. 4.1.1 is an example where plagiarism is detected. In this a paragraph is given as input and another paragraph that is already present in the database. After performing plagiarism check the

detector confirms plagiarism with percentage similarity of 99.74% and cosine similarity is 0.99.

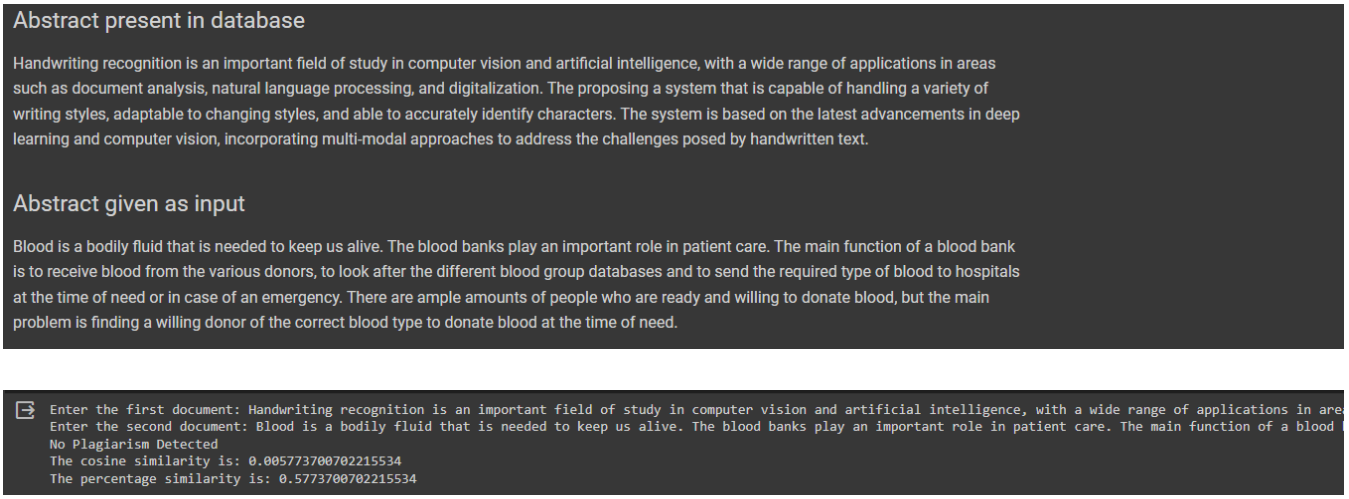
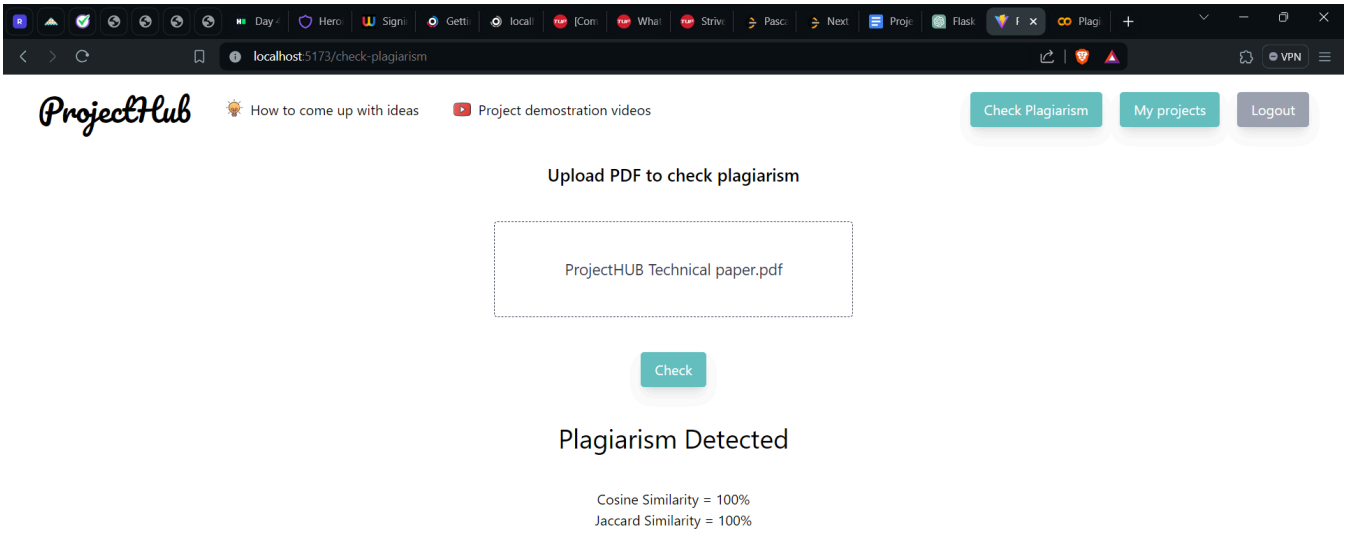


Figure 4.1.2 Result of Cosine Similarity 2

In fig 4.1.2 is an example where plagiarism is not detected. In this a paragraph is given as input and another paragraph that is already present in the database. After performing plagiarism check the detector confirms that there is no plagiarism with percentage similarity of 0.57% and cosine similarity is 0.0057.

Here the threshold for cosine similarity is set as 0.5[17], this means any cosine similarity value that is above 0.5 is termed as plagiarized and any value below this threshold value is termed as not plagiarized and is an original content that can be added forward by the students to make proposal and report.


localhost:5173/check-plagiarism
ProjectHub
How to come up with ideas
Project demostration videos
Check Plagiarism
My projects
Logout
Upload PDF to check plagiarism
ProjectHUB Technical paper.pdf
Check
Plagiarism Detected
Cosine Similarity = 100%
Jaccard Similarity = 100%

Chapter 4

Conclusion

In conclusion, ProjectHub represents a pioneering effort in creating a centralized repository for a diverse array of student projects within a college ecosystem. By harnessing the power of technology and incorporating robust plagiarism detection tools, ProjectHub not only addresses the issue of redundant projects but also fosters a culture of innovation and academic integrity among students.

The platform's integration with plagiarism detection tools ensures that all projects showcased are original and have not been copied from existing sources. This not only upholds the principles of academic integrity but also encourages students to think creatively and propose novel solutions to problems. The ability for students to access and learn from previous projects provides valuable insights, enabling them to build upon existing ideas or devise entirely new ones. The convergence of a centralized project repository and sophisticated Natural Language Processing (NLP) methodologies is meticulously orchestrated. The project showcases a nuanced implementation of ensemble learning, strategically fusing the cosine and Jaccard similarity algorithms to capitalize on their respective advantages. In summary, ProjectHub stands as a symbol of innovation in academic project management, incorporating dynamic NLP techniques. The strategic use of ensemble learning in plagiarism detection reinforces our commitment to precision, fostering a culture of originality and academic honesty within the scholarly community. Positioned as a comprehensive repository for students, educators, and researchers, ProjectHub not only tackles redundancy issues but also sets a high standard for technologically-driven, collaborative educational platforms.

References

[1] Sagar Kulkarni, Dr. Sharvari Govilkar, Dhiraj Amin, “Analysis of Plagiarism Detection Tools and Methods”.

[2] Faouzia Benabbou and Hambi el Mostafa, “A System for Ideas Plagiarism Detection: State of art and proposed approach”, IAES International Journal of Artificial Intelligence (IJ-AI), 2020.

[3] A.C. Santha Sheela, Dr. C. Jayakumar, “Comparative Study of Syntactic Search Engine and Semantic Search Engine: A Survey”, Fifth International Conference on Science Technology Engineering & Management (ICONSTEM) 2019.

[4] Eman Salih Al-Shamery and Hadeel Qasem Gheni, “Plagiarism Detection using Semantic Analysis”, Indian Journal of Science and Technology 2016.

[5] Aparna Babu, Anju S, Archana P Udayan, Giny Mary John and Divya S B, “Online Plagiarism Detection and Result Evaluation using Data Mining and NLP”, International Journal of Engineering Research & Technology (IJERT) 2022.

[6] Mahwish Abid, Muhammad Usman, Muhammad Waleed Ashraf, “Plagiarism Detection Process using Data Mining Techniques”, The International Journal of Engineering & Science(IJES) 2017.

[7] Adil Ahnaf, Hossain Mohammad Mahmudul Hasan, Nabila Sabrin Sworna, Nahid Hossain, “An improved extrinsic monolingual plagiarism detection approach of the Bengali text”, International Journal of Electrical and Computer Engineering (IJECE) 2023.

[8] Hiten Chavan, Mohd. Taufik, Rutuja Kadave, Nikita Chandra, “Plagiarism Detector Using Machine Learning", International Journal of Research in Engineering, Science and Management (IJRESM) 2021.

[9] Nikhil Paymode, Rahul Yadav, Sudarshan Vichare, Suvarna Bhoir, “Online Assignment Plagiarism Detector”, International Journal of Advanced Research in Science, Communication and Technology (IJARSCT) 2021.

[10] Oscar Karnalim, Simon, “Explanation in Code Similarity Investigation”, IEEE ACCESS 2021.

[11] El-Rashidy, M.A., Mohamed, R.G., El-Fishawy, N.A. et al. An effective text plagiarism detection system based on feature selection and SVM techniques. Multimed Tools Appl 83, 2609–2646 (2024). https://doi.org/10.1007/s11042-023-15703-4

[12] Edio da Costa, Vasco Soares Mali, "Tetun Language Plagiarism Detection With Text Mining ApproachUsing N-gram and Jaccard Similarity Coefficient", Timor-Leste Journal of Engineering and Science, Vol.2, Issue.1, pp.11-20, 2021.

[13] Somefun T. E., Awosope C. O. A., Sika C., "Development of a research project repository", TELKOMNIKA Telecommunication, Computing, Electronics and Control 2020.

[14] Ridhima Rajesh Khyalappa, Ankit Verma, Clarence Franklin Fernandes, M. Nitish Reddy, Deepak G, Harish Kumar N, "Challenging Approaches in Project Repository Management System", International Journal for Research in Applied Science & Engineering Technology (IJRASET) 2022

[15] Jasmin Tumulak, Rodel Balingit, Marian Malig-on, "Digital Project Repository: A Community for Students to Collaborate Research Projects".

[16] Heeseok Choi, Jaesoo Kim, "National Repository of Papers based on Open Access", IEEE 2017.

[17] Saeed, A.A.M. and Taqa, A.Y. (2022) 'A proposed approach for plagiarism detection in Article documents,' Sinkron : Jurnal Dan Penelitian Teknik Informatika, 7(2), pp. 568–578. https://doi.org/10.33395/sinkron.v7i2.11381.

[18] Zhang, Xiaodan & Hu, Xiaohua & Zhou, Xiaohua. (2008). A comparative evaluation of different link types on enhancing document clustering. 555-562. 10.1145/1390334.1390429.