Journal of Trends in Computer Science and Smart Technology (ISSN: 2582-4104)
www.irojournals.com/tcsst/

Winnowing Algorithm: A Powerful Tool for Identifying Plagiarism in Assignments

Shiva Shrestha[1], Sandeep Gautam[2], Kiran Sharma[3], Abinay Bhandari[4]

Electronics and Computer, Himalaya College of Engineering, Tribhuvan University, Kathmandu, Nepal

Email: [1]shiv.shres25@gmail.com, [2]macabresndp@gmail.com, [3]keerushar21@gmail.com, [4]bhandariavinay@gmail.com

Abstract

Plagiarism refers to using other ideas or works as their own without giving proper acknowledgment. The act of plagiarism is inappropriate and untrue for many reasons, especially in the academic world. Academicians are aware of this and try to avoid the act of plagiarism by any means necessary. In the present context, the digital way of teaching and learning is in practice which has more chance of plagiarized content. This research provides plagiarism detection features due to the lack of such features in digital-based teaching-learning activities. This proposed system handles the document in text format and uses Winnowing Algorithm for fingerprinting the assignment documents, and the hashing technique chosen for this algorithm is the Rolling Hash function. The similarity value is calculated using Jaccard coefficient. The test results show the combinations of parameters (n-gram, window length, and the base prime number) for the successful implementation of the system. The system successfully detects plagiarism on student assignments. The overall system is developed by using Python Web Framework Django and MySQL as a database.

Keywords: Plagiarism, Winnowing Algorithm, Rolling Hash Function, Jaccard Coefficient, Python.

DOI: https://doi.org/10.36548/jtcsst.2023.2.006
Received: 27.04.2023, received in revised form: 27.05.2023, accepted: 10.06.2023, published: 15.06.2023

1. Introduction

In this modern era, with the advancement of technology and equipment like computers, the availability of others' work, project, and reports is made easy which leads to the result as plagiarism. Plagiarism refers to using others' ideas or works as their own without giving due credit. Plagiarism is considered a harmful behavior within academia that can have adverse effects on educational progress at the higher education level. Consequently, scholars strive to comprehend this academic misconduct. One way they tackle this issue is by offering precise explanations of plagiarism, aiming to prevent any confusion when dealing with plagiarism detection and prevention [1].

With the advancement of technology, most of the work is done online. Moreover, assignment of students is also submitted in electronic forms, as it is easy for both teacher and student. But there is more chance of plagiarism in such e-forms. As the internet is easily accessible to most users, acquiring research, papers, articles, and books are feasible which leads to plagiarism. This has negative impacts on students' studies. So, there must be some way to check for the plagiarism of the data [2]. Plagiarism can be classified into different forms. Some are easy to identify, and some are difficult. Some ways of plagiarism are:

- Patch writing: Using another author's words in your work without giving proper attribution, after rephrasing them.
- Paraphrasing: Rewriting the text in other words.
- Fake citations: Citing information but with incorrect sources.
- Self-plagiarism: Taking your previously completed work and repurposing or reusing it.
- Direct plagiarism: Intentionally incorporating another author's words or ideas into an assignment without acknowledging their source.
- Accidental plagiarism: Unintentionally failing to acknowledge or cite material obtained from external sources due to lack of awareness or oversight. As plagiarism is done differently, it is difficult to detect it manually, so an automated system is required to detect plagiarism. For, there are different ways

by which it can be done automatically and efficiently too. Some of the techniques are, document comparison done by algorithm, method using language-specific structure, etc. [3].

1.1 Problem Statement

As per the present context, most of the teaching and learning activities are in a phase of shifting from their traditional way to a new digital platform. Although there are a lot of positive aspects of this transformation, there occurs huge issues related to the management of assignment systems in digital or softcopy platforms. Students can copy tasks already done by other people and submit them as their own task, this may led to a rise in plagiarized content and unauthorized access to one's knowledge or task which may affect overall teaching-learning activity.

1.2 Objectives

- To find similarities in the content and ensure that it is original.
- To help in protecting the copyright of written content.
- To create a web application that consists of information on students, teachers, and the respective task for which they are assigned.

1.3 Scope and Applications

The system can have more applications in today's world. It will benefit the user in the following perspective:

- For Education and Business: This system can be used by teachers, and students to detect whether their content is plagiarized or not. Moreover, with the help of the system, unauthorized copies of content can be checked which helps in business too.
- To help in protecting the copyright of written content.
- The suggested system's scopes include quick results and complete online functionality.

2. Theoretical Background

2.1 Winnowing

The winnowing algorithm is the document fingerprinting algorithms that can be used to determine whether documents have been copied. Finding similarities between a document and other documents is done using the fingerprint document itself. Whitespace insensitivity, which involves eliminating superfluous characters like punctuation, is one of the prerequisites of the plagiarism algorithm that has been satisfied by the winnowing algorithm [4]. Figure 1 illustrates some of the fundamental steps that are involved in calculating the document fingerprint.

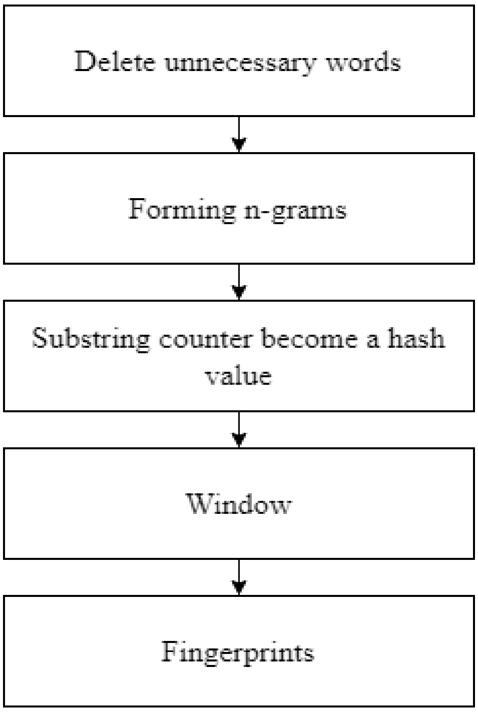

Figure 1. Formulation of Winnowing Algorithm

2.2 Jaccard Coefficient

The Jaccard Coefficient equation, also known as the Jaccard Index, is a similarity assessment method that compares scores based on the intersection and union of calculated fingerprints [5]. The calculation of document similarity is derived from the Jaccard Coefficient equation, as depicted below.

$$Similarity(di, dj) = \frac{|W(di) \cap W(dj)|}{|W(di) \cup W(dj)|} * 100\%$$

Here, *W(di)* represents the fingerprint generated from the first text document, while W(dj) represents the fingerprint generated from the second text document. The values for the Jaccard coefficient lie between 0 and 1, with less than 0.5 indicating less moderate plagiarism and greater than 0.5 indicating extreme plagiarism [6].

3. Requirement Analysis

3.1 Project Requirement

This research is an approach to building a system that distinguishes whether the content submitted by the student is copied from others or not. For this, the machine should have a minimum system requirement of 8 GB RAM and good storage capacity to hold the dataset.

3.2 Feasibility Study

A feasibility study involves whether the project is possible to implement or not. The criteria to judge feasibility are cost requirement, technical acceptance, deliverables, etc. The credibility of a feasibility study for potential investors relies heavily on the perceived objectivity of the study, as it assesses the project's potential for success [7]. Feasibility studies can be divided into various types:

3.2.1 Economic Feasibility

Estimated prices of hardware and software are affordable.

3.2.2 Technical Feasibility

The system is being developed in Python using various algorithms. With the help of some of the libraries and Python’s functionality, all the desired functional requirements for this system can be implemented. The proposed research is technically feasible as it is accurate, reliable, and secure.

3.2.3 Operational Feasibility

Operational feasibility includes an operational analysis of the overall system. The system is operationally feasible because it has a good user interface with better usability and understandability. Also, it aims to fulfill day-to-day user requirements related to checking for plagiarism.

4. System Design

4.1 Use Case Diagram

The proposed research involves three primary users: the administrator, teachers, and students. The administrator assumes overall control of the system's users and is responsible for assigning them to specific departments. Students can utilize the portal to upload their assignments and monitor the status of their work, tracking whether their respective teachers have accepted them. On the other hand, teachers can access all submitted assignments and assess the similarity rates of those assignments already stored in the system's database. The respective roles of these users can be visualized through the use case diagram presented in Figure 2.

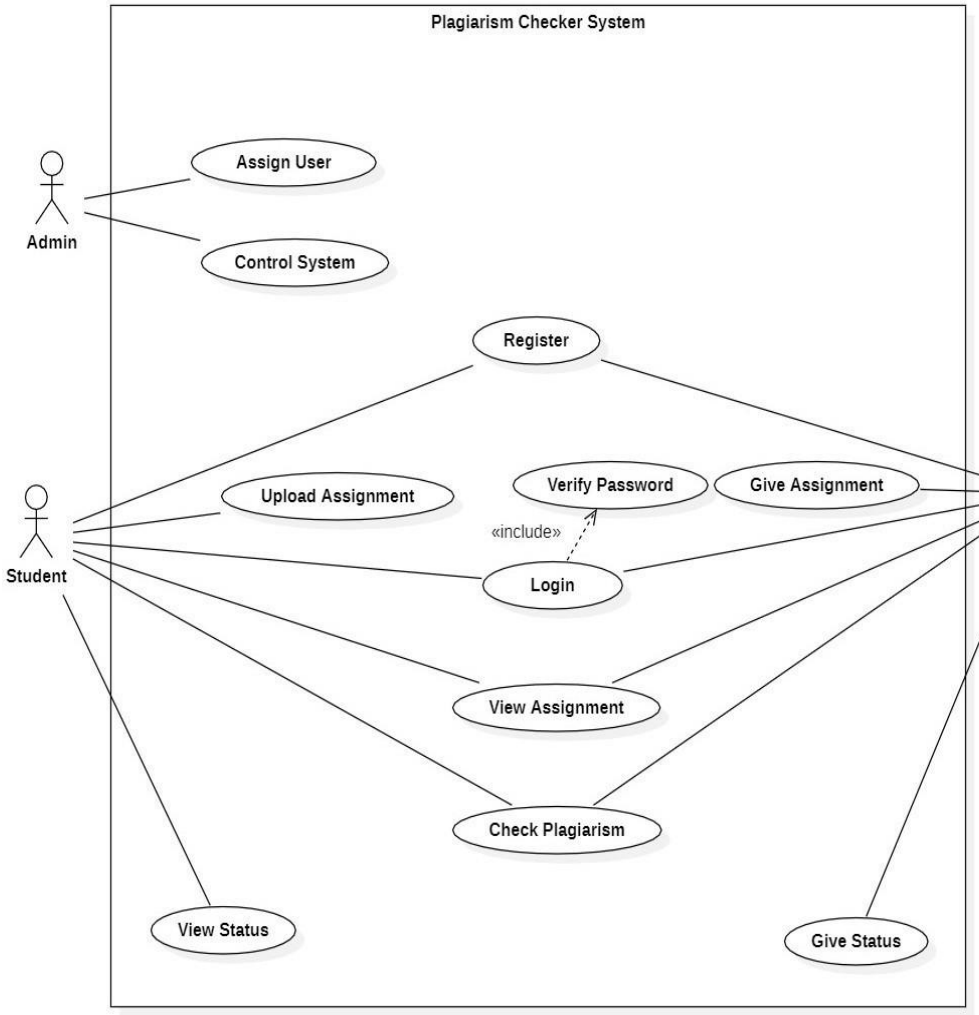

Figure 2. Use Case Diagram

4.2 Class Diagram

The system contains five major objects such as, admin, teacher, student, subject, and assignment. Each class's object has its own attributes and functionalities. For instance, admins have a username, email address, and password as attributes that are used for accessing the system. Admins perform operations like login, assign teachers, modify teachers, add subjects, etc., and have relationships with other objects like teachers, subjects, and students. The blueprints of the system can be understood coherently by a class diagram, as shown in Figure 3.

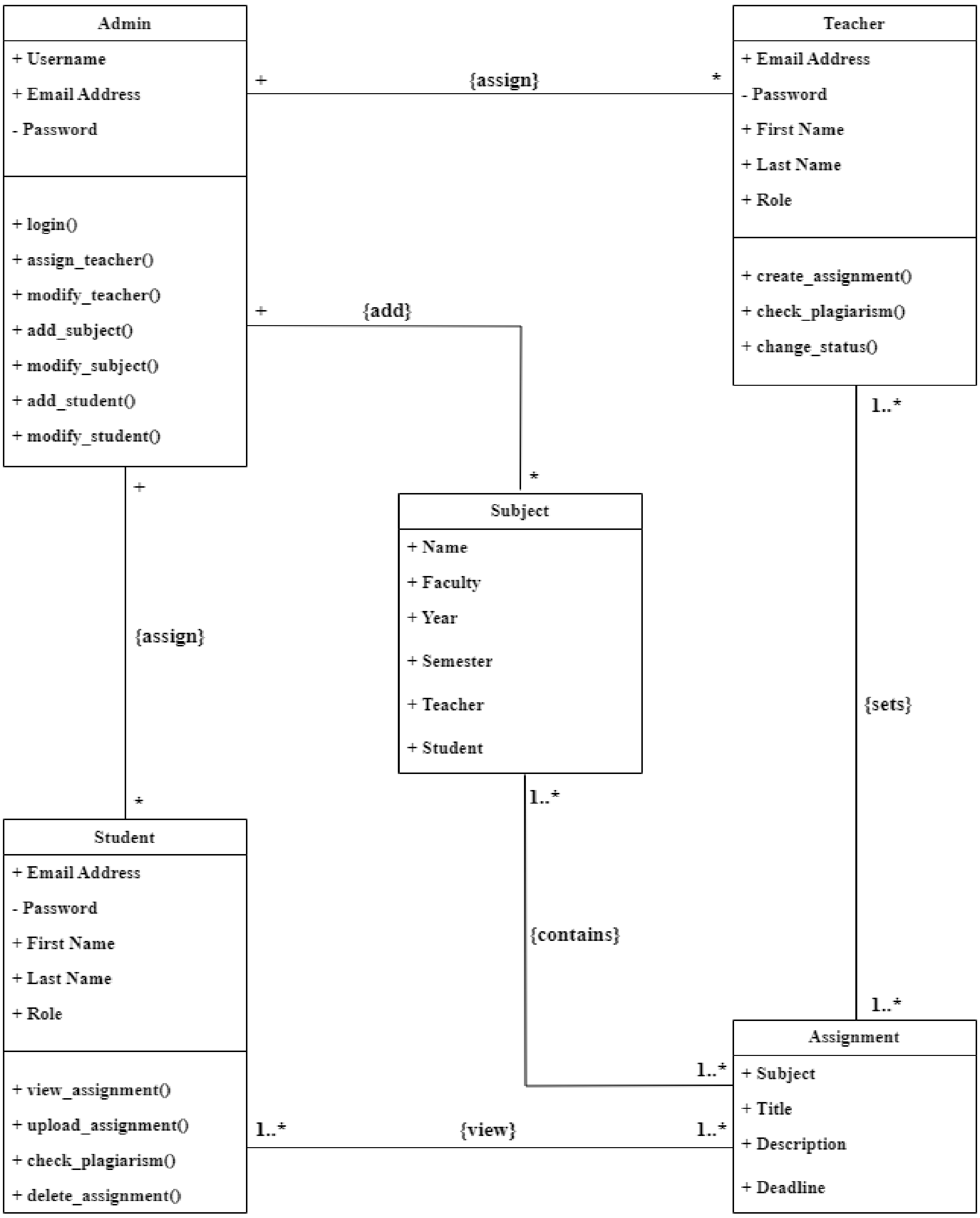

Figure 3. Class Diagram

4.3 Entity Relationship Diagram

The proposed system has six main entities: admin, user, student, subject, teacher, and assignment. Each entity represents different operations to be performed and has its attributes. For example, the user entity represents general users of the system and has attributes including name, email, password, and user ID (uid). Users can have different roles, such as students or teachers, and can perform various actions within the system. An admin assigns users to different roles, and according to the specified roles, teachers can give assignments and students can submit assignments for review. The entities involved in the system facilitate a better understanding of how the system functions and how they are interconnected. The entity relationship diagram shown in Figure 4 provides a visual representation of the relationships and attributes of the entities involved in the system.

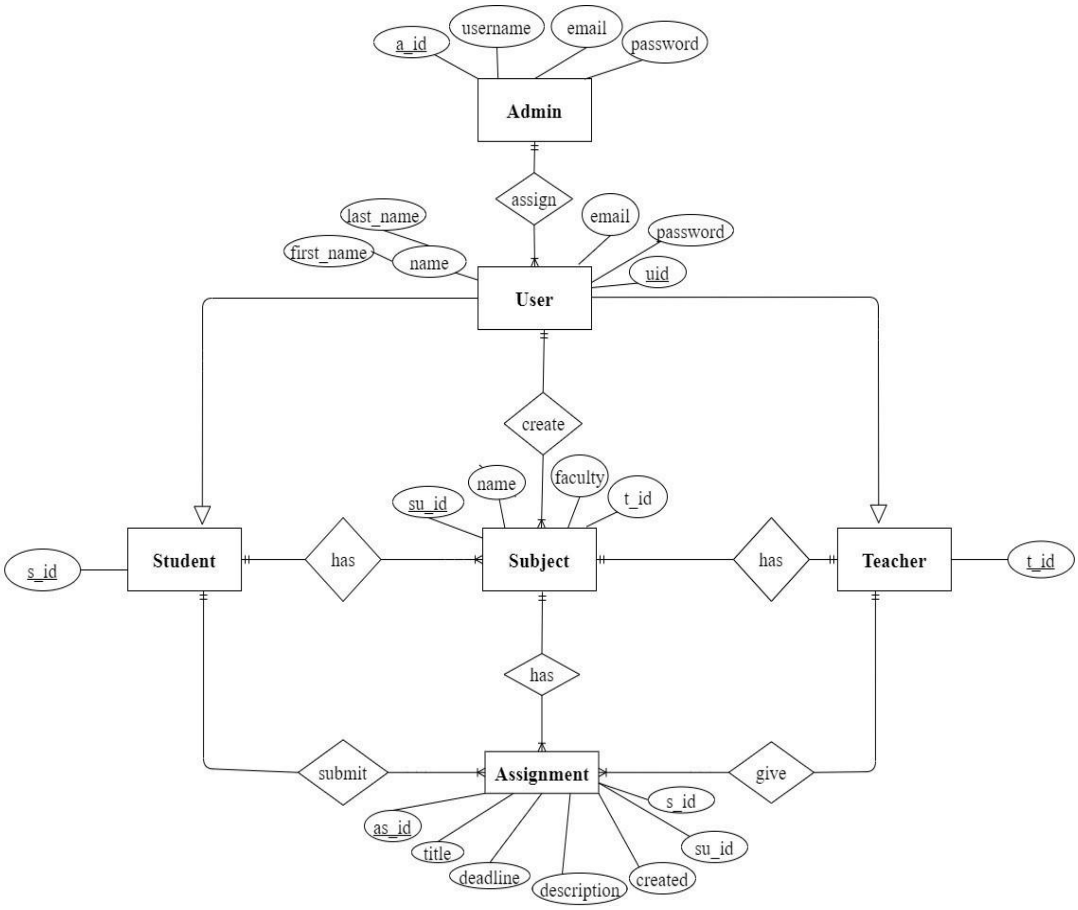

Figure 4. Entity Relationship Diagram

4.4 Flowchart

The system starts with the login screen, where users like admins, students, and teachers can log in. If the user is not registered, then the system pops up a sign-up form from which necessary details like name, email, faculty, and password are set and saved to the database. After registration, the user can log in based on the roles assigned by the admin personnel. Students, as a role, were directed to the student home page and teachers to the teacher home page. The home page for students allows features like uploading, viewing, and checking for plagiarism in self-submitted assignments. On the other side, the teacher's home page allows teachers to create assignments, check similarity rates, and based on the score, accept, or reject them. The overall working pipeline of the project can be understood by the flowchart given in Figure 5.

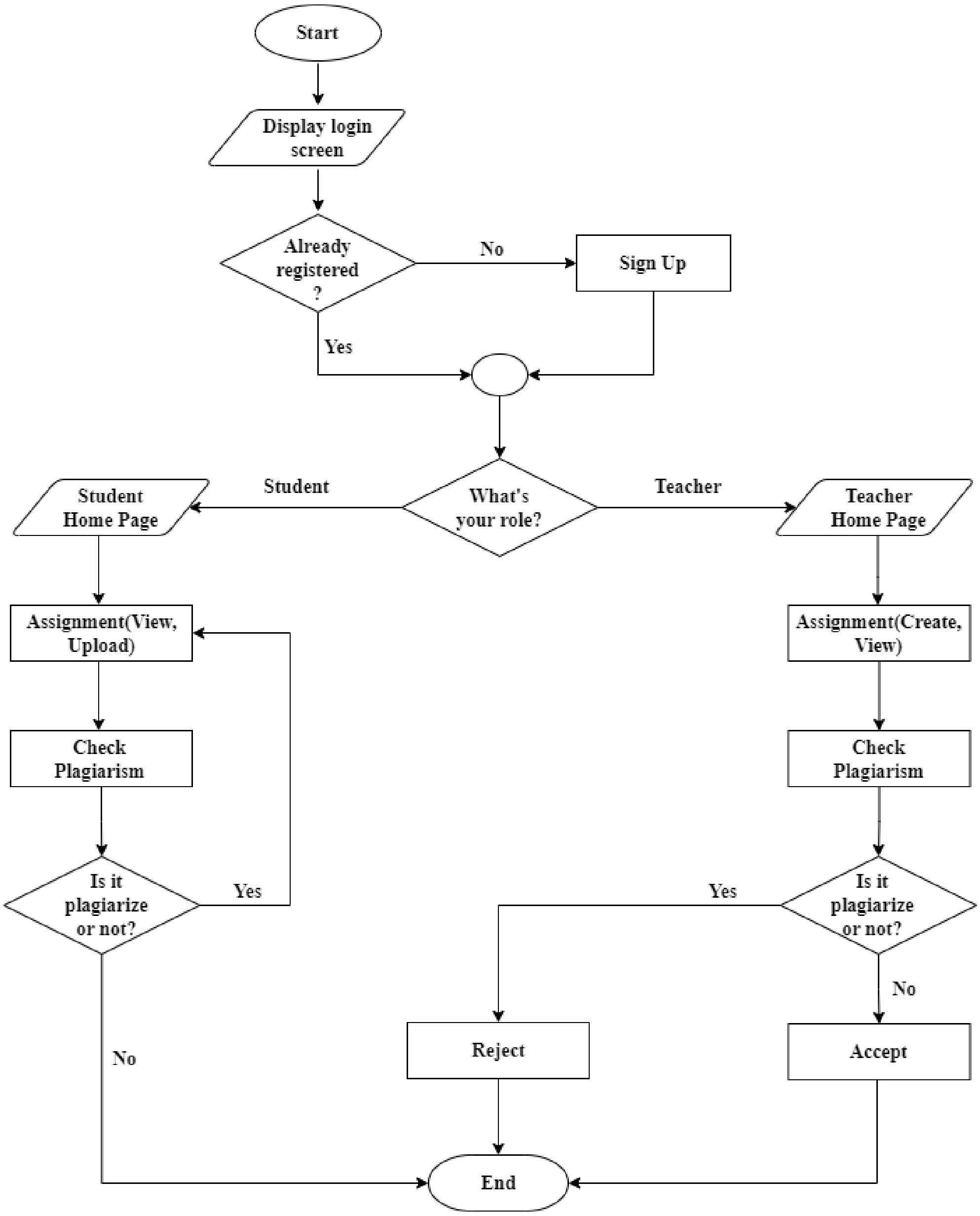

Figure 5. Flowchart

5. Methodology

5.1 System Architecture

The proposed system architecture comprises five main components: input assignment, winnowing process, database, comparison module, and similarity score. The system receives assignments in PDF format, which then undergo a winnowing process. The process begins with the preprocessing of the texts in the PDF, which includes converting text to lowercase and removing digits, stopwords, and punctuation. Sequences of three consecutive characters (tri-grams) are extracted from the preprocessed text, from which a hash value is calculated. The hash values are divided into a specific window size, from which the minimum hash value is selected from each window. The list of selected minimum values gives the fingerprint of the document. The database stores the pre-computed hash values of the documents. The comparison module consists of a similarity measurement tool called the Jaccard Coefficient that compares the newly generated fingerprints in sequential order against the pre-computed fingerprints. The visual representation of the system architecture in which components interact, is shown in Figure 6.

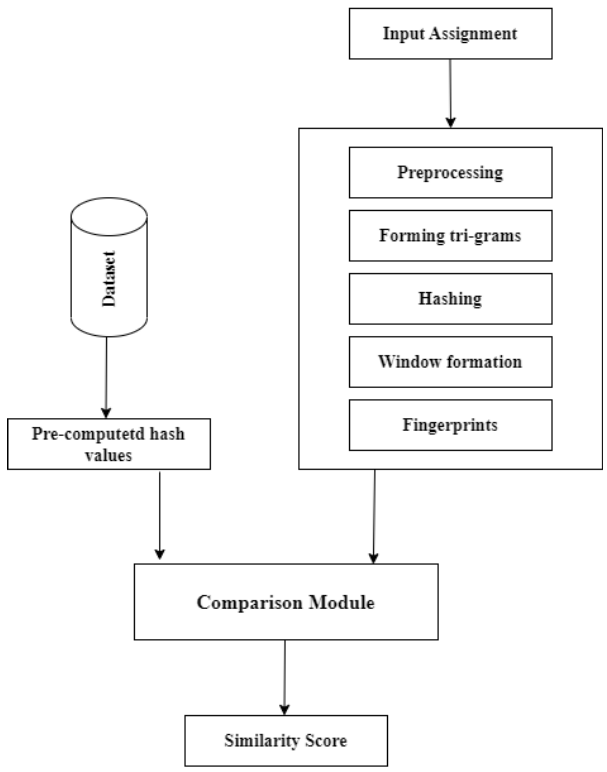

Figure 6. System Architecture

5.2 Working Principle

There are various steps involved in the computation of the project. They are mentioned below:

Collection of data: For plagiarism detection, a collection of data is gathered from diverse sources in electronic format. These sources include academic journals, research papers, online submitted assignments, online publications, and educational repositories. For the proposed system, around 9,000 documents were collected and stored as local data within the system's infrastructure. Initially, the system calculates the fingerprints of all the documents stored in the system storage at once and generates a JSON file with each document's respective file location along with its fingerprint. Whenever new documents are introduced in the datasets, the system calculates fingerprints through a winnowing process periodically and appends the result to the previously created JSON file. It ensures that the dataset is readily available for comparison and analysis, enabling timely and accurate detection of potential instances of plagiarism. By maintaining a comprehensive and up-to-date collection of data, the plagiarism detection system can effectively compare submitted assignments or documents against this dataset, identifying similarities and potential instances of plagiarism. The local data repository forms a crucial component of the system, providing a solid foundation for accurate and reliable plagiarism detection.

Pre-processing: It is the most important step in which the data collected as dataset and data which must be checked for plagiarism are pre-processed. For extracting text from documents, the system uses the Python library ‘pdfplumber’ [8]. After extracting the text, several tasks like converting all the uppercase letters to lowercase letters and removing punctuation marks and spaces are done. Furthermore, the pictures, figures, and numbers are eliminated. These can be accomplished by using the Python library NLTK which supports tokenization, removing numbers and punctuation, eliminating stop words, and stemming and forming n-grams [9].

Constructing of n-grams: ‘n’ number of successive word sequences are said to be n-grams. For easy operation, the preprocessed text is distributed among three successive word sequences called tri-grams [10]. The tri-grams are formed by using three successive letters one after another in the whole text that can be obtained by using Python NLTK library's utility method 'n-grams'.

Hash value generation: Rolling hash is a method of calculating hash values in a sequential manner, where the hash only depends on the substring within a sliding window [11]. Several rolling hash functions have been suggested, and these algorithms maintain a state where each byte is added to the state as it is processed, and later removed from the state after a specific number of other bytes have been processed [12]. Rolling hash technique is used to identify the hash value of thus obtained tri-grams. Rolling hash is given as:

$$H(c_1 \dots c_k) = c_1 * b^{(k-1)} + c_2 * b^{(k-2)} + c_3 * b^{(k-3)} + \cdots + c_{(k-1)} * b + c_k$$

In the given context, the rolling hash equation involves utilizing the ASCII value (c), a base prime number (b), and the number of characters (k).

Window formation: The hash value thus obtained after applying the rolling hash function on each of the tri-grams is subjected to be in several windows having windows size of length four [13]. The smallest value will be chosen from each of the windows, and if more than one smallest value is in a single window then the rightmost value is selected [14].

Similarity measurement: The comparison then proceeds between the fingerprint of the assignment to be checked and the dataset which is stored as the local dataset. The comparison is done by using the Jaccard Coefficient Similarity test which shows similar score values in percentage.

5.3 Tools Used

The plagiarism detection system is developed by using Python as a core programming language and Django as a Python web development framework. The tools that are used are mentioned below:

- Visual Studio Code as code editor
- Web Browser
- MySQL as a database

6. System Testing

Testing is an essential process of evaluating a system or its components to determine if they meet the specified requirements. It involves executing the system to identify any discrepancies, errors, or missing elements compared to the desired requirements. Before implementing the system, a trial run is conducted to eliminate any bugs. Testing plays a crucial role in ensuring the success of a system [15]. Once the system's programs are organized, a test plan is created, and the system is run using a specific set of test data. The results of the test run should align with the expected outcomes. This research work includes several stages of testing, some of which are mentioned below:

6.1 Unit Testing

During the development phase, each module is tested independently to view whether the desired output is achieved or not. By unit testing, the proper functioning of an individual part of the system is verified. One of the test cases is the comparison of fingerprints between two common and two different documents.

6.2 Integration Testing

After unit testing is accomplished by proper functioning, each individual module is integrated and a compact system is formed, then the overall system is tested to identify whether there is any fault in the integration or not.

6.3 Test Cases for Functional Testing

6.3.1 For User Registration

Table 1. Test Case for User Registration

S.No.	Test Case	Expected Outcome	Actual Outcome	Remarks
1	Enter each field with valid data	Successful registration	Same as expected	Validated
2	Enter data with some mandatory field empty	Unsuccessful registration	Same as expected	Validated

6.3.2 For Document Comparison

Table 2. Test Case for Document Comparison

S.No.	Test Case	Expected Outcome	Actual Outcome	Remarks
1	Compare two same documents	Show document as maximum plagiarized document	Same as expected	Validated
2	Compare two different documents	Show document as minimum plagiarized document	Same as expected	Validated

6.3.3 For User Role Test

Table 3. Test Case for User Role

S.No.	Test Case	Expected Outcome	Actual Outcome	Remarks
1	Teacher trying to create an assignment session	Teacher must be able to create an assignment session	Same as expected	Validated
2	Student trying to create an assignment session	Students must not be allowed to create assignment session	Same as expected	Validated
3	Student trying to submit their task	Student must be allowed to submit their task	Same as expected	Validated
4	Teacher role on task submitted by student	Teacher must be allowed to either accept or reject the task	Same as expected	Validated

6.4 Implementation Testing

There are various attributes used during the computation of this work such as size n-grams, window length, and base prime number. This attribute plays an important role in the accuracy of the algorithm. For the selection of appropriate values for those attributes, several test cases are implemented which are shown in Tables 4 & 5.

Table 4. Configuration Number for Winnowing Algorithm

Configuration Number	n-gram	Window length	Base prime number
1	3	4	5
2	5	8	11
3	5	5	17
4	3	4	23

Table 5. Test Cases for Verification of Configuration Number

Original text	Compared text	Plagiarism Checker X similarity score	C1	C2	C3	C4
A	A*	54.9%	58%	57%	57%	57%
B	B*	60.2%	60%	61%	61%	61%
C	C*	68%	70%	69%	69%	70%
D	D*	91%	89%	89%	89%	90%

Thus, from Tables 4 & 5, the error percentage rate in Configuration 1 is greater than other configuration values. The remaining configuration values C2, C3, and C4 show approximately the same error percentage rates. Configuration C4 has greater base prime numbers than others, and this algorithm uses Configuration 4 as the best attribute.

7. Result Analysis and Discussion

The main attributes used in the winnowing algorithm are n-gram (n), window size (w), and base prime number (b). Selecting the appropriate values for those parameters that give the minimum error percentage rate with the maximum similarity score, various tests were performed on different documents. From the checked documents with this system along with plagiarism software X, attributes n-gram, window size, and base prime number with values 3, 4, and 23 gave the maximum similarity score compared to other attribute values like ($n = 3$, $w = 4$, and $b = 5$), ($n = 5$, $w = 5$, $b = 11$), and so on. Hence, for the maximum outcome or accuracy of the system, the main attribute values were taken as $n = 3$, $w = 4$, and $b = 23$.

The overall system was tested for any faults or errors, and the result was analyzed. Firstly, the user interface of the system was interactive for a better user experience. Another part is the registration of users; as shown in Figure 7, there is a certain mandatory field in the registration form. If all the mandatory fields are entered, then the registration is successful; otherwise, the registration is invalid or not completed. Moreover, each field consists of varied forms of attributes; all the fields should be filled as per the desired attributes for successful registration.

After successful registration, users, i.e., students or teachers, have different roles to which they are assigned. Teachers can give assignments on the subject that they are assigned, as presented in Figure 8. Students can view the assignment and submit it within the given deadline; during submission, they are allowed to check whether the submitted document is plagiarized or not, as shown in Figure 9. After the student submits their task, the teacher can view the student's task and check whether it is copied or not. Based on the result, the teacher can decide whether to accept or reject each student's assignment individually. Also, the system provides ad hoc comparison of the two documents on the go, as shown in Figure 10, from which users can get their similarity rates on the go and make the necessary changes.

Thus, after the completion of this work, the main problem of "management of digital assignments," which was the objective of the research, is managed, and with the help of this system, the use of plagiarized documents can be minimized.

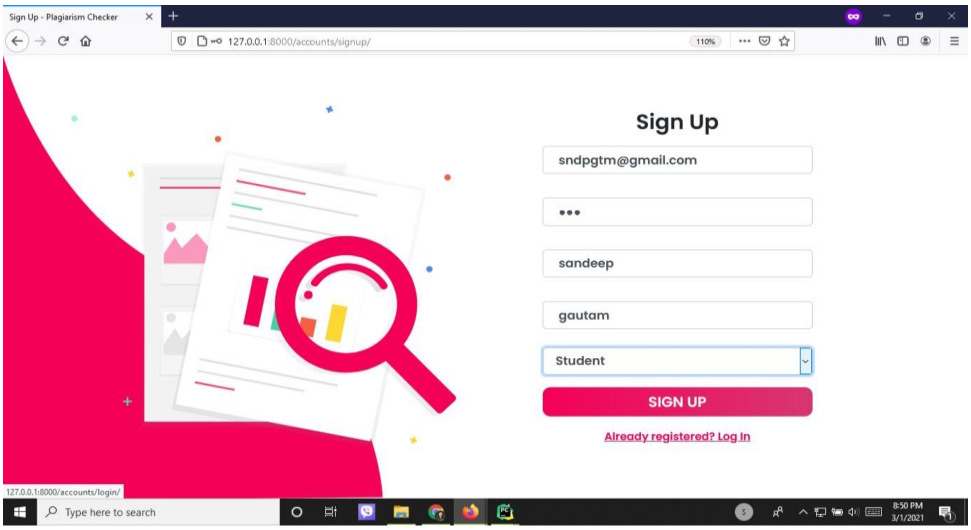

Figure 7. User Registration Form

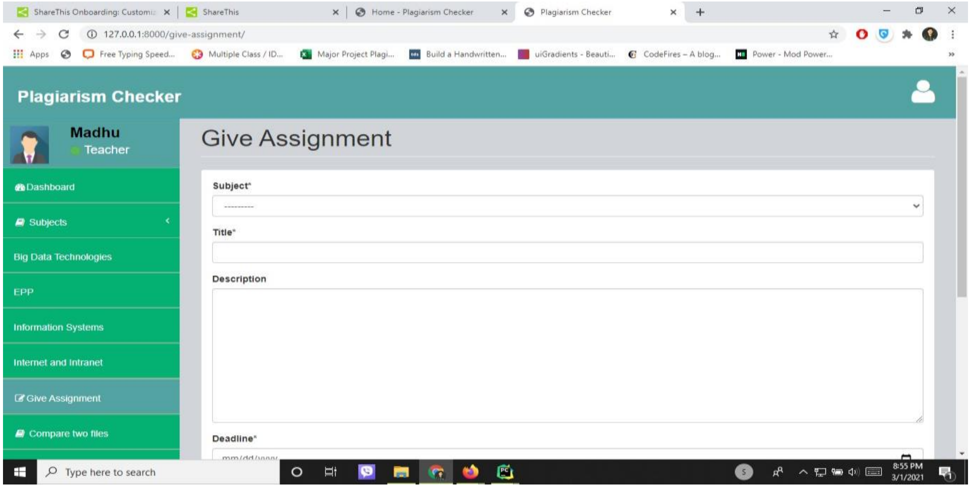

Figure 8. Assignment Portal

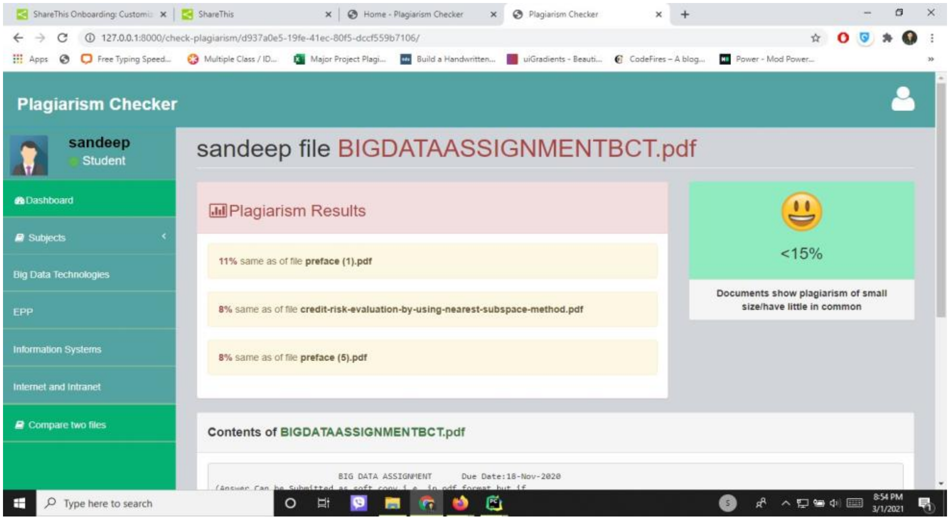

Figure 9. Similarity Score of Document

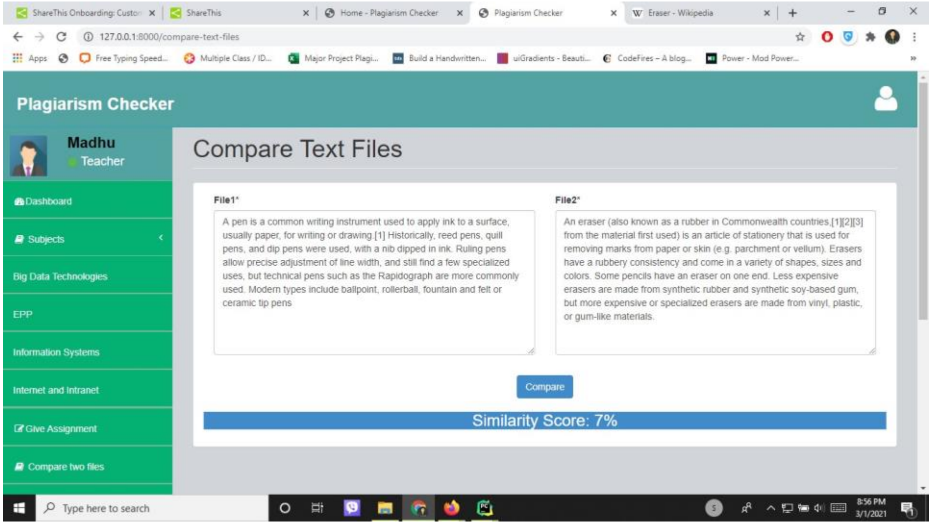

Figure 10. Ad hoc File Comparison

8. Conclusion and Future Enhancement

With the completion of this work, the main aim of the research is achieved, that is easy and efficient system to check plagiarized documents. This web application integrates various functionalities to provide the best user experience for managing digital assignment systems by making users work an efficient, time-saving, and less tedious job.

8.1 Limitation

The research assists well to record the problem related to document plagiarism. However, this work has some limitations such as, it only shows the plagiarized content in percentage beside the actual plagiarized content.

8.2 Future Enhancement

To further enhance the capability of this web application, the actual content or phrases that are copied from original source are to be shown in the system.

Declarations

Conflict of Interest

The authors declare that they have no conflict of interest.

Funding

Not applicable.

References

[1] Akbar, A. (2018). Defining plagiarism: A literature review. Ethical Lingua: Journal of Language Teaching and Literature, 5(1), 31-38.

[2] Park, C. (2003). In other (people's) words: Plagiarism by university students--literature and lessons. Assessment & evaluation in higher education, 28(5), 471-488.

[3] 7 Common Types of Plagiarism, With Examples. (2022, February 15). 7 Common Types of Plagiarism, With Examples | Grammarly Blog. https://www.grammarly.com/blog/types-of-plagiarism/

[4] Hasan, E. G., Wicaksana, A., & Hansun, S. (2018, June). The implementation of winnowing algorithm for plagiarism detection in Moodle-based e-learning. In 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS) (pp. 321-325). IEEE.

[5] Jaccard index - Wikipedia. (2011, March 1). Jaccard Index - Wikipedia. https://en.wikipedia.org/wiki/Jaccard_index

[6] Duan, X., Wang, M., & Mu, J. (2017). A plagiarism detection algorithm based on extended winnowing. In MATEC Web of Conferences (Vol. 128, p. 02019). EDP Sciences.

[7] Bowen, D. J., Kreuter, M., Spring, B., Cofta-Woerpel, L., Linnan, L., Weiner, D., ... & Fernandez, M. (2009). How we design feasibility studies. American journal of preventive medicine, 36(5), 452-457.

[8] Koning, B. (2022). Extracting Sections From PDF-Formatted CTI Reports (Bachelor's thesis, University of Twente).

[9] Kannan, S., Gurusamy, V., Vijayarani, S., Ilamathi, J., Nithya, M., Kannan, S., & Gurusamy, V. (2014). Preprocessing techniques for text mining. International Journal of Computer Science & Communication Networks, 5(1), 7-16.

[10] Sidorov, G. (2013). Non-linear construction of n-grams in computational linguistics. México: Sociedad Mexicana de Inteligencia Artificial.

[11] H. Jiang and S. -J. Lin, "A Rolling Hash Algorithm and the Implementation to LZ4 Data Compression," in IEEE Access, vol. 8, pp. 35529-35534, 2020, doi: 10.1109/ACCESS.2020.2974489.

[12] J. Kornblum, "Identifying almost identical files using context triggered piecewise hashing," Digital investigation, vol. 3, pp. 91–97, 2006.

[13] E. G. Hasan, A. Wicaksana and S. Hansun, "The Implementation of Winnowing Algorithm for Plagiarism Detection in Moodle-based E-learning," 2018 IEEE/ACIS 17th International Conference on Computer and Information Science (ICIS), Singapore, 2018, pp. 321-325, doi: 10.1109/ICIS.2018.8466429.

[14] Schleimer, S., Wilkerson, D. S., & Aiken, A. (2003, June). Winnowing: local algorithms for document fingerprinting. In Proceedings of the 2003 ACM SIGMOD international conference on Management of data (pp. 76-85).

[15] Sharp, I., Yu, K. (2019). System Testing. In: Wireless Positioning: Principles and Practice. Navigation: Science and Technology. Springer, Singapore. https://doi.org/10.1007/978-981-10-8791-2_11.

Author's Biography

Shiva Shrestha is currently working as a software engineer at Verisk Analytics. He received his bachelor's in computer engineering from Tribhuvan University in 2021. His area of research includes natural language processing, cloud computing, and decoding machine learning algorithms.

Sandeep Gautam is pursuing his second semester of MBA-IT in the Faculty of Management, Tribhuvan University, Nepal. He received a Bachelor of Computer Engineering from Tribhuvan University in 2021. He is currently engaged in SOMTUVmag.

Kiran Sharma pursued his bachelor's degree in computer engineering from Tribhuvan University. He began his career as a software engineer, working on a service-based project for a US-based company. He also works as an assistant lecturer at reputed IT colleges in Nepal.

Abinay Bhandari received a bachelor's degree in computer engineering from Tribhuvan University in 2021. He works as a software engineer at a service-based company. His research areas include big data, cloud computing, and machine learning.