Plagiarism Detection System

Ashish Jain
Computer Science and Engineering

Anmol Kumar Pandey
Computer Science and Engineering

Aniket Saini
Computer Science and Engineering

ABSTRACT

Plagiarism is frequently referred to as "literary theft" and "academic dishonesty" in the literature, and as it is a rising problem, it is important to be knowledgeable about the subject in order to avoid it and uphold ethical ideals. The proliferation of knowledge on the internet and in digital libraries has made plagiarism one of the most significant problems facing colleges, universities, and research sectors. Finding student-written materials or journals is incredibly simple thanks to the internet and sophisticated search engines. Plagiarism, then, is a widespread issue that affects many facets of our lives. It is crucial to note right now that identifying plagiarism is a difficult task. However, the need for technology is necessary for detecting plagiarism easily. Despite the fact that search engines like Google are available, it would be annoying to search phrases repeatedly to examine related resources on the internet. Because plagiarism may be automatically detected and highlighted, implementing a plagiarism detection system which will speed up the process of identifying plagiarism. All that is required is for the user to upload the paper to the detection system. Consequently, this document suggests that we have developed a software program for users and successfully tested it for plagiarism detection in student assignments. It is an efficient web-enabled system for detecting plagiarism.

General Terms

Your general terms must be any term which can be used for general classification of the submitted material such as Pattern Recognition, Security, Algorithms et. al.

Keywords

Plagiarism Detection, text-matching software, machine learning

1. INTRODUCTION

Plagiarism is described as “The practise of taking someone else's work or idea and putting it on as one’s own”. According to the report, it was found that Every third student uses the internet to research and collect information their Assignments and plagiarized them, and 59% of high school students acknowledged to cheating. The COVID-19 epidemic has resulted in an exponential increase in online education and testing. Teachers should do thorough quality checks on students' assignments to reduce plagiarism as soon as possible. Every document must be checked for plagiarism, which takes time and is tedious. There are many plagiarism detecting programmes on the market to assist teachers, assessors, and researchers. Plagiarism can take many various forms, and it can be quite demotivating for both teachers and students in schools. If plagiarism is not effectively addressed, perpetrators may benefit unfairly, such as receiving higher grades for their assignments with less effort. In numerous applications, including file manager, plagiarism detection, and copyright infringement, the identification of plagiarized documents plays a crucial role. Plagiarism is commonly committed by duplicating entire paper or only a portion of it, changing the language used to express the same idea, borrowing ideas from others, or citing erroneous or nonexistent sources. Other forms of plagiarism include creative plagiarism, in which diverse mediums like photos and videos are utilize to show other people's work without proper attribution, and translated plagiarism, an example of translated plagiarism involves translating the content and using it without mentioning the original source.

2. LITERATURE REVIEW

Martial, a Roman poet, used it for the first term in a literary setting perhaps around 80 AD. Poets were required at the time to be able to recite important works by other authors. However, Martial was shocked to learn that Fidentinus, a rival poet, was quoting his works and claiming authorship. Martial decided to answer. One of the biggest trends in academia is the move away from online and copy/paste plagiarism and toward essay mills and plagiarism for hire. In certain nations, like the UK and New Zealand, it has even been the subject of legislation.Our [2] reviewed the current state of the art for identifying academic plagiarism, provided plagiarism detection tools, and compiled analyses of how well they catch plagiarism. The drawbacks of text-based plagiarism detection techniques were discussed, and it was recommended that future research concentrate on semantic analysis techniques that also take into account non - textual document aspects, such academic citations. While concentrating on source code PD, Agarwal and Sharma [3] also provided a general overview of text content plagiarism detection techniques. Source code PD and PD for text are related technologically, and many PD for text methods can also be used to identify plagiarism in source code [4]. For text documents entered in the PAN competitions, Kanjirangat and Gupta [5] described plagiarism detection techniques and contrasted four algorithms.

3. DESIGN AND IMPLEMENTATION

Plagiarism detection system is implemented by using a python language. For detecting the percentage of plagiarism system compare text of both files and if similarity between sentences and word arrangement found then it will added in a plagiarism and plagiarism percentage increase. For design gui of this system use html and css . This project include many parameters.

3.1 Libraries

Re - The functions in this module allow you to determine whether a given string matches a given regular expression (or RE), which describes a set of strings that match it (or if a given regular expression matches a particular string, which comes down to the same thing).

3.2 For reading files

```
Text = open(root_file,”r”)

text = Text.read()

pattern_file = open(plagiarised_file,”r”).read()
```

3.3 To convert text in lower case

Pat[j].lower()==text[i].lower()

3.4

Pat[j].lower()==text[i].lower()

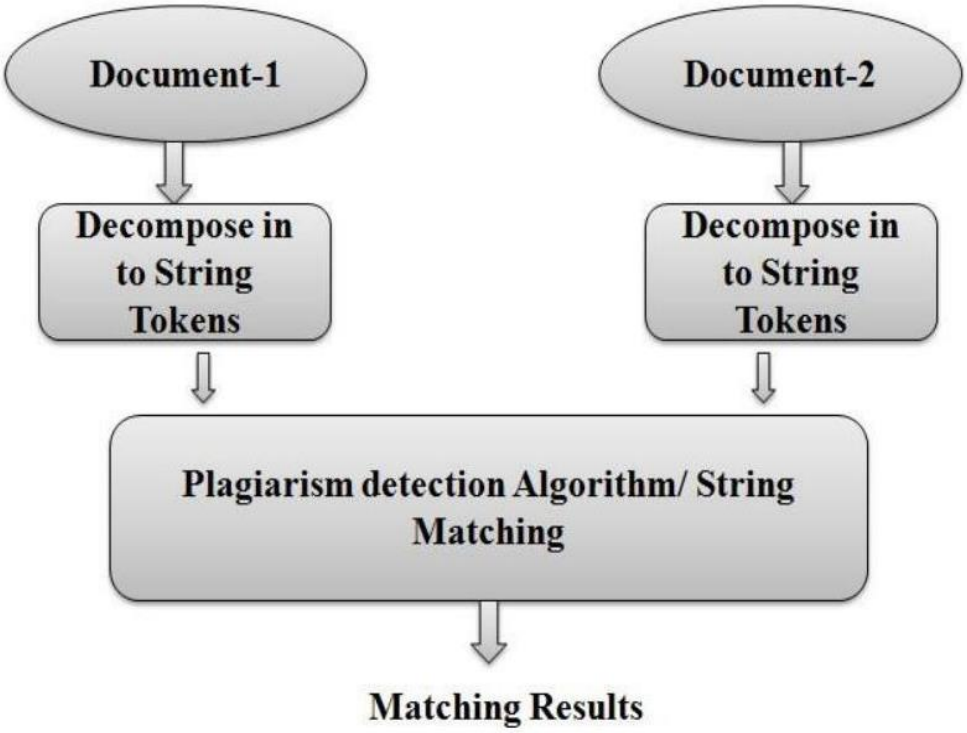
Document-1
Document-2
Decompose in to String Tokens
Decompose in to String Tokens
Plagiarism detection Algorithm/ String Matching
Matching Results

Fig 1: System flow chart

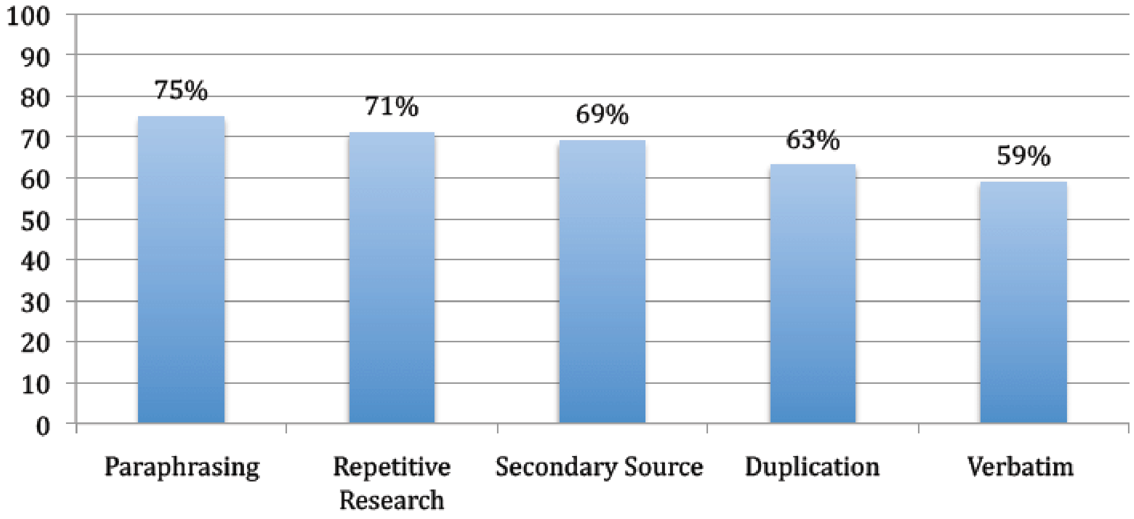

Fig 2: Common plagiarism types

4. PROPOSED METHODOLOGY OF WORK

Our method is built to find plagiarism when two text documents are copied and pasted. The project's primary objective is to combine several small tactics for improved outcomes. To be sure of the outcomes in this case, we employ the idea of substring matching. Two methods for pattern matching were discovered. The Knuth-Morris-Pratt algorithm and the Rabin Karp algorithm. The rolling hash function [4] and substring matching are two advantages of the Rabin Karp algorithm that are present at the pre-processing stage.The hash of the pattern, let's say h (p), matches the source's h(s), and the string match is then verified. The complexity of this algorithm is O (m+n) in the best- case scenario and O in the worst (mn). In this case, we merely use the KMP method and not the Rabin Karp algorithm. The degenerating characteristic of the pattern is used by the KMP algorithm. We are already familiar with the preceding words of the window if there is a mismatch (following a few matches). We avoid looking for those character combinations again. The worst-case complexity for this algorithm is O(n).The number of matches and the duration for the match were the two metrics used to examine algorithms. A collection of strings S1 in one document can be matched to another set of strings S2 in another document using the Knuth-Morris-Pratt (KMP) string matching method. The algorithm is based on forward pattern matching. This algorithm pattern matches each string of S1 to each string of S2 thus providing accurate match but requiring more time to provide such results. Hence a parallel KMP can be used to process more numbers strings at a time.

5. EXPERIMENTAL RESULT

The output window will show the matching percentage of both the files. If the match is more than a certain threshold (default value 60%), an additional line will get printed to the console - "The input file seems to have been copied. x% of its content is identical to that of file F." (F - name of the file with matching percentage >= threshold, x - the matching percentage.)

Example: If two files are selected which contains the biography of famous actor Brad Pitt named as bradpitt.txt and bradpittbio.txt and checked these files on this application . These files contain the content about actors details but there may be similarity if both these files so after the execution the application shows the amount of similarity in the documents in the form of percentage.As an output it shows that respective file is plagiarised by 34.66% and as it doesn't cross the threshold value 40% so this file is not plagiarised.

6. CONCLUSION

For assignments in which some concealed types of copying can be identified, such as sentence structure modification and synonym replacement, we have tested experimentally a prototype of a plagiarism detector. We have outlined its structure. Also,Incorporating a plagiarism detection approach into educational systems is essential and necessary, to sum up. There is a critical necessity to safeguard the submitted student work's intellectual property due to the global expansion of e - learning. This study clarified the problem and suggested a productive paradigm for document submission that incorporates both a system and a procedure for detecting plagiarism.The results show that the technique is extremely effective and allows for plagiarism checks. The instructor can take a report from one of his or her courses and show it to the students, which details the portions that were copied from each assignment that was turned in.A lot of systems are effective at identifying and discouraging plagiarism, but they also aid in teaching students the value of originality. The future development will concentrate on improving and expanding the capabilities of this tool.

7. REFERENCES

[1] E. Walter, Cambridge Advanced Learner's Dictionary with CD-ROM. Cambridge university press, 2008.

[2] "2012 report card," Nov 2021.

[3] E. Marais, U. Minnaar, and D. Argles, "Plagiarism in e-learning systems: Identifying and solving the problem for practical assignments," in Sixth IEEE International Conference on Advanced Learning Technologies (ICALT'06), pp. 822– 824, IEEE, 2006.

[4] Q. Li, S. Li, S. Zhang, J. Hu, and J. Hu, "A review of text corpus-based tourism big data mining," Applied Sciences, vol. 9, no. 16, p. 3300, 2019.

[5] V. Liu and J. R. Curran, "Web text corpus for natural language processing," in 11th Conference of the European Chapter of the Association for Computational Linguistics, pp. 233–240, 2006.

[6] Y. Kumar, D. Mahata, S. Aggarwal, A. Chugh, R. Maheshwari, and R. R. Shah, "Bhaav-a text corpus for emotion analysis from hindi stories," arXiv preprint arXiv:1910.04073, 2019.

[7] R. R. Naik, M. B. Landge, and C. N. Mahender, "Development of marathi text corpus for plagiarism detection in the marathi language," corpus, vol. 6, p. 340, 2011.

[8] S. P. Green, "Plagiarism, norms, and the limits of theft law: Some observations on the use of criminal sanctions in enforcing intellectual property rights," Hastings LJ, vol. 54, p. 167, 2002.

[9] H. A. Chowdhury and D. K. Bhattacharyya, "Plagiarism: Taxonomy, tools and detection techniques," arXiv preprint arXiv:1801.06323, 2018.

[10] A. H. Osman, N. Salim, and A. Abuobieda, "Survey of text plagiarism detection," Computer Engineering and Applications Journal, vol. 1, no. 1, pp. 37–45, 2012.

[11] G. Navarro, "A guided tour to approximate string matching," ACM computing surveys (CSUR), vol. 33, no. 1, pp. 31–88, 2001.

[12] S. S. Skiena, The algorithm design manual, vol. 2. Springer, 1998.

[13] S. Zhang, Y. Hu, and G. Bian, "Research on string similarity algorithm based on levenshtein distance," in 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), pp. 2247–2251, IEEE, 2017.

[14] Ding, W. and Marchionini, G. 1997 A Study on Video Browsing Strategies. Technical Report. University of Maryland at College Park. Fröhlich, B. and Plate, J. 2000. The cubic mouse: a new device for three-dimensional input.