Online Assignment Plagiarism Checking Using Data Mining and NLP

Taresh Bokade[1], Tejas Chede[2], Dhanashri Kuwar[3], Prof. Rasika Shintre[4]

[1-3]Student, Computer Engineering, SIGCE, Navi Mumbai, Maharashtra, India
[4]Asst. Professor, Computer Engineering, Smt. Indira Gandhi College of Engineering, Navi Mumbai, Maharashtra, India

---***---

Abstract - *Plagiarism is a big problem in academics and it can be a big problem in every department in education sector. Students plagiarize in different areas: homework assignments, essays, projects, etc. Academics know that information can support valuable learning experiences, but these experiences are diminished when students plagiarise by copying assignments and getting credit for work they have not done. In this project we are going to develop a system for plagiarism detection in which whenever a student submits an assignment it detects that it is plagiarized or not by comparing with other students assignments. For this we will use data mining algorithms and natural language processing to get proposed output.*

Key Words: Data Mining, NLP, WordNet, SCAM, KMP

1. INTRODUCTION

Plagiarism is defined as to take or theft some work and present it has one's own work. This grammar and plagiarism checker system is used to analyze the plagiarism data. Plagiarism affects the education quality of the students and thereby reduce the economic status of the country. Plagiarism is done by paraphrased works and the similarities between keywords and verbatim overlaps, change of sentences from one form to other form, which could be identified using WordNet etc. This plagiarism detector measures the similar text that matches and detects plagiarism. Internet has changed the student's life and also has changed their learning style. It allows the students to get deeper in the approach towards learning and making their task easier. Many methods are employed in detecting plagiarism.

Usually plagiarism detection is done using text mining method. In this plagiarism checker software, user can register with their basic registration details and create a valid login id and password. By using login id and password, students can login into their personal accounts. After that students can upload assignment file, which will further divide into content and reference link. This web application will process the content, visit each reference link, and scan the content of that webpage to match the original content. Also, students can view the history of their previous documents. Teacher also able to check the grammar mistakes on the content and symantical plagiarism.

1.1 OBJECTIVES

1. To compare the assignment with all other submitted assignment for plagiarism. Example-If the batch having 100 students then a single assignment is checked with all other 99 assignments.
2. To check with syntactical and symantical approach.
3. Exceptional changes like diagram and tables will be checked for plagiarism.
4. Plagiarism detection report will be generated.
5. To add missing citations or rewrite your text.

1.2 SCOPE

In the last few years, Plagiarism detection became so essential topic in researches area. Plagiarism can be involve in different fields research papers, art area and program code. In digitalized future, everything will be in online mode nothing will be there on pen-paper basis. So the plagiarism checking application will be definitely most helpful in the future.

1.3 FEATURES

1. This system can be viewed by students and teachers also.
2. History is available for both students and teachers.
3. Symantical plagiarism checking is also possible.
4. Fast processing of assignments.

2. LITERATURE SURVEY

[1] Juan et al.created a tool called beagle which uses some collusion method to identify plagiarism. This software measures the similar text that matches and detects plagiarism. Internet has changed the students life and also has changed their learning style. It allows the student to deeper the approach towards learning.Many methods are employed in detecting plagiarism. Usually plagiarism is done using text mining method.

[3]Steve et al. proposed an automatic system to detect plagiarism. This system uses neural network techniques to create a feature based plagiarism detector and to

measure the relevance of each feature in that available assessment. This paper solely focus on two different aspects namely copy-paste type and paraphrasing plagiarism types only. The results were compared with commercially available online software "Article checker".

[4]Nathaniel et al.defines plagiarism as a serious problem that infringes copyrighted documents/materials. They proposed a novel plagiarism-detection method called as SimPAD. The purpose of this method is to establish the similarities between two documents by comparing sentence by sentence. Experiments say that SimPAD detects plagiarized documents more accurate that out performs existing plagiarism-detection approaches.

[5]Jinan et al. focused on the educational context and faced similar challenges. They describe on how to check the plagiarism cases. In addition they planned to build learning communities-communities of students, instructors, administration, faculty and staff and all collaborating and constructing strong relationships that provide the foundation for students to achieve their goals with greater success. They provided seamless integration with legacy and other applications in some easy, modifiable, and reusable way. Learning portal may provide a support tool for these learning system. This paper gives the software to detect the plagiarism from java student assignments.

[6]Hermann et al.say that plagiarise is to robe credit of another person's work. He describes the first attempt to detect the plagiarised segments in a text employing statistical language models and perplexity. The experiments were carried out on two specialised an literary corpora. The two specialised works contained the original documents and part-of speech and stemmed versions. They detected the plagiarism on these documents and the results were verified.

[7]Francisco et al.say that laboratory work assignments are very important for computer science learning. Study says that over the last 12 years 400 students copy the same work in the same year in solving their assignment. Thus they developed a plagiarism detection tool. This tool had the full toolset for helping in the management of the laboratory work assignment. They used four similarity criteria to measure the similarities between two assignments. Their paper described how the tool and the experience of using them over the last 12 years in four different programming assignment.

3. PROBLEM STATEMENT

Finding plagiarized parts of a assignment is very slow work for teachers. Even with a limited number of texts it relies on the teacher's ability to read and remember every submission. As the process of finding plagiarized parts in assignment is based on the teacher's ability to remember all that he or she has read, the results may be incomplete. Some clear cases of copy and paste may easily be overlooked. And since the workload cannot be shared between multiple assistants. Thus we are introducing system for checking Plagiarism in assignments.

4. PROPOSED SYSTEM

In proposed system, we are going to develop a system to detect the plagiarism in the academic assignment which will help to stop copying the assignment of other student and will improve the quality of education and also will help to improve personal skills of student and student can also check the grammar from the assignment. In this system plagiarism detector measures the similar text that matches and detects plagiarism. As well symantical checking will be also done with respect to assignment. For detecting the plagiarism we will use data mining algorithm and natural language processing.

4.1 FLOWCHART

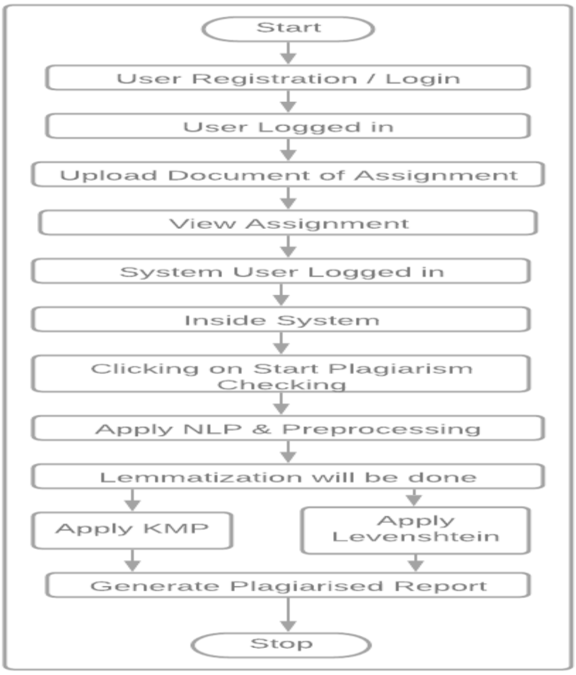
Start
User Registration / Login
User Logged in
Upload Document of Assignment
View Assignment
System User Logged in
Inside System
Clicking on Start Plagiarism Checking
Apply NLP & Preprocessing
Lemmatization will be done
Apply KMP
Apply Levenshtein
Generate Plagiarised Report
Stop

Figure-1: Flowchart

Working Flow:

Collection of assignments: All the assignments or documents will be collected in electronic format. So that plagiarism can be detected efficiently.

Pre-processing: Pre-processing is a major step in the process in which all the assignments are converted into a appropriate format. All these assignments collected must be in the same format. Numbers, figure values, pictures and all those things which are not from a-z group should be excluded from the documents.

Classification: Text classification should be performed to extract and separate the parts of a sentence into alternative words. With the help of this key words from a sentence can be found.

Text analysis: Further, the data will be passed through the text analyzing step. This process can be repeated, sometimes, according to the need. Moreover different text analyzing techniques can be used according to the nature of text and aims of the institutes.

Similarity measures: Further in the process, comparison is performed upon the sequence of tri-grams created from the processed documents, with the help of sequences comparing methods.

Clustering the plagiarized data: Clusters are created from the similar tri-grams to calculate the similarity score. Clusters will help in the calculations and will accelerate the process.

Similarity score: Similarity score will be calculated through the clustering of the similar tri-grams. Similarity will be calculated in the form of percentage. High value of percentage depicts the high similarity score.

4.2 ALGORITHMS

1. **Rabin-Karp algorithm:**

It is a search algorithm that searches for a substring pattern in a text using hashing. It is very effective for multi-pattern matching words. The accuracy level can be adjusted based on this feature. The hash function is a function that determines the feature value of a particular syllable fraction. It converts each string into a number, called a hash value. Rabin-Karp algorithm determines hash value based on the same word.

2. KMP:

The technique is to look for the pattern in the text in a left-to-right order. Just like the brute force algorithm, but the shifting in KMP is more intelligently than the brute force algorithm. If a mismatch occurs between the text T and pattern P at P[j], the most shifting can be done to the pattern to avoid wasteful comparisons is the largest prefix of P[1 .. j-1] that is a suffix of P[1 .. j-1]

Suppose: j = mismatch position in P[] k = position before the mismatch (k = j-1) The border function, or also called failure function, b(k) is defined as the size of the largest prefix of P[1..k] that is also a suffix of P[1..k] [4] . If a mismatch occurs at P[j] (i.e. P[j] != T[i]), then k = j-1; j = b(k) + 1; //obtain the new j.Consider an example such there are string T "abacaabaccabacabaabb" and pattern P "abacab", find if there is any match with P in T. The illustration of KMP algorithm will : The mismatch occurs on P[j] where j = 6. The p' is now "abaca". Suffixes from p' are { "a", "ca", "aca", "baca" } while prefixes from P are { "a", "ab", "aba", "abac", "abaca" }. Thus, the longest suffix on p' which is also the longest prefix on P is "a" then shift the pattern as much as: length(p') – length("a") = 5 – 1 = 4. The matching and shifting is performed until exact match is found.

3. SCAM:

This presents a Stanford Copy Analysis Mechanism (SCAM) based on word occurrence frequency. It"s mainly maintains registered document which are used for copy detection. A vector of words with its frequency is used to compare with vectors in the database.

Step1:Pre-processing Stage:

Separation: In this step, the text of input document is isolated from the references mentioned therein. Separating the references from the text can be manually or programmatically.

Tokenization

Step1: Declare String array text[], text2[], Declare String line. Initialize Integer C1_text, C2_text2, sum_text, sum2_ text to 0

Step2: Set line = in.readLine(); // to fetch line from file

Step3: Do WHILE line is not equal to null set text[]=line.split(" "); // split the line based on space increment sum_text; // sum = length of array text[]. ENDWHILE; Set next line by : line = in.readLine(); // fetch the next line Until (end of ile). // Now, all the lines are in array text[].

Step4: WHILE C1_text < sum_text Set text2[C2_text2]=text[C1_text].replaceAll("[\\W]", ""); // delete the delimiters

Increment C1_text

ENDWHILE;

C2_text2=sum2_text2; // sum2= length of array text2

Step5: Print text2[C2_text2] // text2[]= individual words without delimiters.

Stop Words Removing

Stop words are words that repeated frequently in the English language, but do not carry any information. These words may be kind of pronouns, conjunctions and prepositions. he output of this stage is a text free of stop word, initially puts all letters in lower case. Stop words that removed from the text. The output of pre-processing stage is a text ready to check against semantic plagiarism.

Step2 : Document Disciplinary

Before detect the semantic plagiarism, a process of identifying the specialist of document is done to detect plagiarism only for documents that fall within the specialty of computer science, while the documents with other disciplines will not subject to plagiarism detection.

Word Frequency

Ater pre-processing stage, he occurrence of each word in the input document will computed according to how many times it appears in document.

Descending Order

Frequencies that found in the previous step will arranged in descending order.

N Specification

At this stage, N was determined within the program that representing the highest frequencies will be taken from the total number of it.

Word (N)

Words that represent the highest (N) frequencies will take a side to reveal the correlation of source document with the computer science fields.

Decision Making

Finally, the fields that are related to this document will be displayed and continue working.

Step3: Semantic Plagiarism Detection

Then, to help detecting semantic plagiarism, we propose to use semantic similarity between documents based on information extracting techniques. Semantic plagiarism will be detect based on WordNet.

Text

If the document passed from the threshold of specialty test of computer science, the text will be taken once again to complement this work.

WordNet

After taking the text that was specified in the previous step, to determine the extent of the semantic plagiarism, synonyms for each word is to ind using WordNet. Every word in the specfied text will be extracted its synonyms. hense synonyms will be considered as appearance of the word itself when used to detect plagiarism.

WordNet Expansion

At this stage, WordNet expansion has been proposed by speciic words doesn't exist in its dictionary.

Documents of Database

At this stage, the documents stored in the database will be withdrawn one after another, for these documents, the text is taken entirely not just a specific text in it, to the possibility of plagiarize a text exist in different places of database document and put it in another place of source document, these places may be abstract, results.

5. CONCLUSION

Plagiarism detection is essential for protecting the written work. It is concluded that all institutues and and teachers should be aware of plagiarism and anti-plagiarism softwares. We have designed a simple method which assists us with the detection of instances of plagiarism in assignment of school and college students. Our scheme is easy to adapt for the large variety of programming languages in use, and is sufficiently robust to be highly effective in an educational environment. While having a detection rate as good as other more complex software, it presents its report as a simple graph, enabling large numbers of assignments to be checked quickly and efficiently. By using data mining algorithm and NLP it will provides straightforward documentation which can be used as clear and convincing evidence should a suspected instance of plagiarism be disputed.

ACKNOWLEDGEMENT

The success and final outcome of this research required a lot of guidance and assistance and we are extremely privileged to have got this all. All that we have done is only due to such supervision and assistance and we would not forget to thank them.

We respect and thank Prof. Rasika Shintre, for providing us insight and expertise that greatly assisted the research. We are extremely thankful to her for providing such a nice support and guidance.

REFERENCES

[1] "Software metrics and plagiarism detection," J. Syst. Software, vol. 13, pp. 131– 138, 1990.

[2] M. J. Wise, "Detection of similarities in student programs: YAP'ing may be preferable to Plague'ing," ACM SIGCSE Bull., vol. 24, no. 1, pp. 268–271, 1992..

[3] A. Islam and D. Inkpen, Semantic text similarity using corpus-based word similarity and string similarity, ACM Transactions on Knowledge Discovery from Data, vol. 2, no. 2, pp. 125, Jan. 2008.

[4] U. Bandara and G. Wijayarathna, “A Machine Learning Based Tool for Source Code Plagiarism Detection,” International Journal of Machine Learning and Computing, pp. 337–343, 2011.

[5] Eman Salih Al-Shamery and Hadeel Qasem Gheni. Plagiarism detection using semantic analysis.Indian Journal of Science Tech.

[6] Alexander Budanitsky and Graeme Hirst. Semantic distance in wordnet: An experimental, Application - oriented evaluation of five measures. University of Toronto- Toronto, Ontario, Canada.

[7] A. Anguita, A. Beghelli, and W. Creixell, Automatic cross-language plagiarism detection, 2011 7th International Conference on Natural Language Processing and Knowledge Engineering, 2011.

BIOGRAPHIES

Taresh Vishnuji Bokade is pursuing Bachelor degree (B.E.) in Computer Engineering from Smt. Indira Gandhi College of Engineering, Navi Mumbai.

Tejas Balu Chede is pursuing Bachelor degree (B.E.) in Computer Engineering from Smt. Indira Gandhi College of Engineering, Navi Mumbai.

Dhanashri Vijay Kuwar is pursuing Bachelor degree (B.E.) in Computer Engineering from Smt. Indira Gandhi College of Engineering, Navi Mumbai.

Prof. Rasika Shintre, Obtained the Bachelor degree (B.E. Computer) in the year 2011 from Ramrao Adik Institute of Technology (RAIT), Nerul and Master degree (M.E. Computer) from Bharati Vidyapeeth College of Engineering, Navi Mumbai. She is Asst. Professor in Smt. Indira Gandhi College of Engineering having about 8 years of experience. Her area of interest includes Data mining & information retrieval.