Effective Detection of Plagiarism using Semantic Technology

Sukanya S Gaikwad

Research Scholar
Department of Computer Science,
Gulbarga University, Kalaburagi, India

***Abstract*— Plagiarism of digital documents seems a serious problem in today's era. Plagiarism refers to the use of someone's data, language and writing without proper acknowledgment of the original source. Plagiarism of another author's original work is one of the biggest problems in publishing, science, and education. Plagiarism can be of different types. This paper presents a different approach for measuring semantic similarity between words and their meanings. Existing systems are based on the traditional approach. For detecting plagiarism, traditional methods focus on text matching according to keywords but fail to detect intelligent plagiarism using semantic web. We have suggested new strategies for detecting the plagiarism in the user document using the semantic web. In paper we have proposed architecture and algorithms to better detection of copy case using semantic search, it can improve the performance of copy case detection system. It analyzes the user document. After the implementation of this technique, the accuracy of plagiarism detection system will surely increase.**

***Index Terms*— Plagiarism, plagiarism detection, Semantic Web.**

I. INTRODUCTION

The problem of plagiarism or copy case is increasing very rapidly because of digital era of resources available on World Wide Web (WWW). Plagiarism of digital documents seems a serious problem in today's era. Plagiarism refers to the use of someone's data, language and writing without proper acknowledgment of the original source. Plagiarism of another author's original work is one of the biggest problems in publishing, science, and education. Plagiarism in text documents can be in several forms like plagiarized text may be copied one-to-one, passages may be modified to a greater or lesser extent or they may be translated or it is act of claiming to be author of information that actually someone else wrote. The plagiarism can be defined as "the unauthorized use or close imitation of the ideas and language of someone else".So the focus of this paper is to give a plagiarism detection technique using semantic technology that will better catch the plagiarism. Till now several techniques for plagiarism analysis have been proposed. In this paper we have presented one technology that addresses this issue. We have suggested new strategies for detecting the plagiarism in the user document using the semantic web. In paper we have proposed architecture and algorithms to better detection of copy case using semantic search, it can improve the performance of copy case detection system. After the implementation of this technique, the accuracy of plagiarism detection system will surely increase.

II. RELATED WORK

Protection of digital documents from illegal copy has received much attention recently. Most of techniques for copy case detection are based on ideas of substring matching. In paper Gusfield, D. Substring matching approach basically identifies maximum matches in pairs of strings, which will be used as plagiarism indicators later but this is a traditional technique which is limited to the good accuracy.

In paper Fullam K., and Park, J. and paper SI, A., Leong, H.V., and Lau , R.W.H keyword similarity mechanism is used. In this paper the idea was to weight topic and to compare them to the keywords of other documents.

In paper Geoffrey R. Whale the author have proposed graph based performance measures technique for code plagiarism detection.

In paper Broder AZ et al, author has proposed the copy detection method. The new method was able to detect document overlap based string matching but it was not a good idea because it cannot find partial sentence copy. So the accuracy was not good.

The same thing is done by U. Manber and G. Myers. Su_x arrays. This paper was also giving the idea of plagiarism detection technique but it was limited to the good accuracy.

In our paper we have proposed architecture and algorithms to better detection of copy case using semantic search, it can improve the performance of copy case detection system.

III. PROPOSED ARCHITECTURE

The proposed architecture of plagiarism detection technique is shown in Figure 1. The proposed technique semantically compares the user's document with web using wordnet. A text is considered to be a sequence of words each of which carries useful information. Fig. 1 shows the whole procedure for computing the sentence similarity between two candidate sentences. The organization of architecture is explained below:

• *Query Document*: The user uploads the document to check the plagiarism in his document.

• *Decomposition*: This phase decomposes the document in sentences to match properly the data from the ontology.

• *Semantic comparison plagiarism detection*: This phase compares the user document with web to check the plagiarism in his document.

• *Wordnet*: Wordnet is used for getting the synonyms of words.

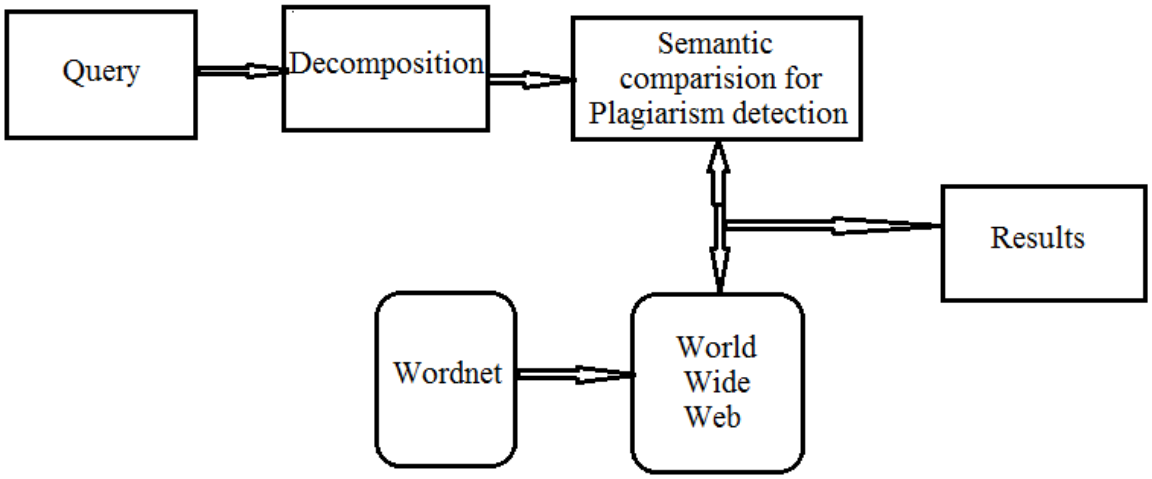

Figure 1: Proposed Architecture of Semantic plagiarism detection.

• *World Wide Web*: It is a system of interlinked hypertext documents .Using the web browser; we can view web pages that may contain text, images and other multimedia. Using the www we can use the web resources to compare user's document.

• *Results*: After the scanning of the document the result will be send to the user with the highlighted lines in which plagiarism was detected and it will also show the percentage of copied work in that particular document. A deterministic view of example of proposed plagiarism detection technique is shown in the figure 2 and figure 3 . The example shows that there are two types of plagiarism detected in the document. One type of plagiarism detected when user replaces the words with its synonyms and the second type of plagiarism occurs when user adjusts the lines.

Original: World Wide Web is a **vast** resource of **data growing** continuously. Nowadays, it **becomes increasingly hard** for **users** to **retrieve useful data** due to the continually **rapid** growth in **data volume**.

Plagiarized: WWW is a **huge** resource of **information increasing** continuously. Now, it is very **difficult** for **client** to **fetch needed information** due to the **fast** growth in **information size**.

Plagiarism Found: Words are replaced by its synonyms.

Green Color: Green color is showing the original document's words which are replaced by Plagiarized document.

Red Color: Yellow color is showing the replaced words from the original document semantically.

Figure 2: Example scenario of intelligent plagiarism extraction technique based on word's semantic

Original: **The first weakness is** that **web users cannot express their search intention accurately or properly using several keywords**. So **most of the time the exactly matched results do not satisfy the web users**.

Plagiarized: **The web users cannot express their search intention accurately or properly using several keywords is the first weakness** . **So the exactly matched results do not satisfy the web users most of the time**.

Plagiarism Found: words are re-arranged.

Yellow color: Starting four words from first sentence of original document re-arranged and placed to end of the sentence.

Blue Color: Plagiarized document has copied the words in the starting of the first sentences.

Pink color: Second sentence of the original document is placed to the end of the sentence in plagiarizes document.

Purple: Plagiarized document has copied the words in the starting of the second sentence.

Figure 3: Example scenario of intelligent plagiarism extraction technique based on word's re-arrangement

IV. PROPOSED ALGORITHM

checkPlagiarism() : The function takes the document to be checked for plagiarism as input. The document is then decomposed into sentences. Each sentence is then matched with the documents available on the web. For the first time, the sentences are checked for the exact match. The result is stored in a boolean variable match. If match is false, the sentence is rebuild using synonyms for each word (different combinations) for semantically checking for match. The sentence is rechecked for match, if the match is again false, it is checked if the sentence comes in a category of large sentence by checking its length (>10 words). If a sentence is large, it is broken into words and then all the words of the sentence are compared to the words of each sentence of the document. Finally, if the match is true in any of the case above, C[i] is made true which tells about the position of the sentence in the array which is copied and M is incremented which is the count for the number of sentences which are found copied. Match ratio is calculated by dividing M by N where N is the total number of sentences in the document and stored into Match_ratio. The percentage of document which is copied is returned as output. The sentences S[i] are highlighted corresponding to which C[i] is true.

Input: Document for checking plagiarism.
Output: Percentage of document copied from web

```
1. checkPlagiarism ()
2. { Boolean match false;
3. C[]  //Boolean Array that stores true at indices
corresponding to copied sentences
4. M=0; //Number of matched sentences
5. d getUserDocument();
6. S[N] decomposeDocument(d);
7. for i=0 to N do
8. match  sentence_match_keyword(S[i]);
9. if(match==false)
10. then
11. Match sentence_match_synonyms(S[i]);
12. if(match==false)
13. then
14. if(S[i].length>10) // length = no. of words
15. then
16. Match words_rearrange_and_match(S[i]);
17. if(match==true)
18. then
19. C[i]=true;
20. M++;
21. Match_ratio=(M/N);
22. Print Match_ratio*100;
23. //Highlight sentences that are copied
24. }
```

Figure 4: Proposed algorithm for checking the plagiarism semantically.

Input: Sentence for the document
Output: true/false to tell whether sentence is plagiarised

```
1. sentence_match_keyword (sentence)
2. {
3. for every sentence
4. words<-split(“ “,sentence);
5. for(i=0;i<words.length;i++)
6. {
7. st<-words[i];
8. for(j=0;j<words.length;j++)
9. if(i!=j)
10. st<-st+” “+words[j];
11. //fetch the suspected pages and count lines
N in suspected page
12. Foreach line t in suspected page
13. {
14. if(t==st)
15. p++;
16. }
17. }
18. if(p/words.length*100>8)
19. return true;
20. else
21. return false;
22. }
```

Figure 5: Proposed plagiarism detection algorithm for checking the suspected sentences.

sentence_match_keyword() : This function takes the sentences one by one, breaks the sentence into individual words and then form all the possible combinations of those words to form different possible sentences and checks if they match to the line in the suspected page/resource. On every match the value of a counter is increased. If this value percentage is greater than 8% then it would return true else false stating the chances of plagiarism.
sentence_match_synonyms() : This function works the same way as the above defined function only addition in this function is that semantic meaning of each word is also used to frame the sentences and then checks it with the suspected page/resource returning true of false.

V. .CONCLUSION

The digital document is being replicated across the server. Many times it happens that people makes the near copies of the original document. In this paper we have proposed architecture and algorithm that can detect the plagiarism using the semantic technology. This approach is better than previous methods because earlier paper was focusing on the keyword based plagiarism detection technique but this technique is detecting the plagiarism in a better way using the semantic technology.

REFERENCES

1. E.K. Park, D.Y. Ra, and M.G. Jang, "Techniques for Improving Web Retrieval Effectiveness," Information Processing and Management, vol. 41, no. 5, pp. 1207-1223, 2005.

2. Bao Jun-Peng, Shen Jun-Yi, Liu Xiao-Dong, Liu Hai-Yan, Zhang Xiao-Di. "Document Copy Detection Based On Kernel Method," In 2003 International Conference on Natural Language Processing and Knowledge Engineering Proceedings.

3. Narayanan Shivakumar, Hector Garcia-Molina "Building a Scalable and Accurate Copy Detection Mechanism" 1st ACM International Conference on Digital Libraries (DL'96), March. 1996. pp. 160-168.

4. S. Hannabuss. Contested texts: issues of plagiarism. Library Management, 22(6/7):311–318, 2001.

5. GUSFIELD, D. (1997): Algorithms on Strings, Trees, and Sequences: Computer Science and Science and Computational Biology. Cambridge University Press.

6. FULLAM, K., and Park, J. (2002). Improvements for scalable and accurate plagiarism detection in digital documents.

7. SI, A., LEONG, H.V., and LAU, R.W.H. (1997): Check: a document plagiarism detection system. SAC '97, 70–77, New York, NY, USA. ACM Press.

8. Geoffrey R. Whale. Identification of Program Similarity in Large Populations. The Computer Journal, 33(2):140–146, 1990. doi: 10.1093/comjnl/33.2.140.