Automatic Plagiarism Detection Using Similarity Analysis

Shanmugasundaram Hariharan
Department of Computer Science and Engineering, TRP Engineering College, India

Abstract: *Plagiarism involves reproducing the existing information in modified format or sometimes the original document as it is. This is quiet common among students, researchers and academicians. This has made some strong influence on research community and awareness among academic peoples to prevent such a kind of malpractice. Though there exits some commercial tools to detect plagiarism, still plagiarism is tricky and quiet challenging task due to abundant information available online. Commercially existing software adopt methods like paraphrasing, sentence matching or keyword matching. Such techniques are not too good in identifying the plagiarized contents effectively. However this paper focuses its attention on identifying some key parameters that would help to identify plagiarism in a better manner. The results seem to be promising and have further scope in detecting the plagiarism.*

Keywords: *Plagiarism detection, similarity measures, information extraction, text matching.*

Received February 1, 2010; accepted October 24, 2010

1. Introduction

Plagiarism is defined as the use or close imitation of the language and thoughts of another author and the representation of them as one's own original work [7]. Plagiarism comes from a latin verb that means, “to kidnap” If we plagiarize it means that we are kidnapping and stealing others hard work and intellectual property, which is a form of academic and public dishonesty [15]. By the use of synonyms, plagiarism can be done. Therefore, they are difficult to recognize by the commercial software. Plagiarism affects the education quality of the students and there by 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 another form [12], which could be identified using wordnet [1] etc.

Academics know that student valuable learning experience is supported with the help of information, but by the use of plagiarism these experience get demolished. Regarding project based activities for academics it is believed that plagiarism cannot be done easily but still some students try to plagiarize by copying the work done by the other students which is difficult for the faculty to find out. Juan *et al*. [10] 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 and making their task easier. Some students take superficial approach in learning which makes their task easier and thus student tend to copy the work done by others. Detecting plagiarism in a mass of students is difficult and also they are expensive too. Many methods are employed in detecting plagiarism. Usually plagiarism is done using text mining method.

Alan *et al*. [2] created a computer algorithm for plagiarism detection. They proposed an algorithm for detecting plagiarism. The ultimate goal of this software is that to reduce plagiarism. Steve *et al*. [14] 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 the assessment. Students are becoming more comfortable with cheating. Study says that 70% of the students do their work using plagiarism. 40% of the student just copy and paste the work assigned to them. There are many existing software tool. In common practice these plagiarism methods are hard to identify. Some of these methods includes copying of textual information, paraphrasing (representing same content in different words), using content without reference to original work, artistic (presenting same work using different forms), code plagiarism (using program codes without permission or reference), misinformation of references (adding reference to incorrect or non existing source) [6]. To solve such types of plagiarism an enhanced version with mix of algorithm is required to reduce dishonesty indulged to academic environments. 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”. We have identified some important aspects that would identify the plagiarism in a better way compared to the existing tools.

The rest of the paper is organized as follows. Section 2 explains the related works carried out, section 3 briefs the experimental setup. Finally, section 4 gives the conclusion and future improvements.

2. Related Work

Allan *et al*. [4] presented a framework for plagiarism detection. The growth of internet, with abundant information online makes the problem even worse. The authors have found four different ways to approach plagiarism detection. They decided to follow exhaustive searching and took the middle ground method rather than exhaustively or randomly searching sentences in a student paper on the internet. They found the possible sources of borrowed ideas. Plagiarism can be detected with intelligent selection of sentences from papers, which can also be found using internet search engines. They aimed this to develop freeware that for any instructor or teaching assistant can use to detect plagiarism in their classes.

Nathaniel *et al*. [11] defines plagiarism as a serious problem that infringes copyrighted documents/materials. They say that plagiarism is increased now a days due to the publications in online. 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 and outperforms existing plagiarism-detection approaches.

Jinan *et al*. [9] 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 all collaborating and constructing strong relationships that provide the foundation for students to achieve their goals with greater success. They also promoted information sharing. 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. But building and modifying learning portal is not a easy task. This paper gives the software to detect the plagiarism from java student assignments.

Hermann *et al*. [6] say that plagiarise is to robe credit of another person's work. According to the authors, text plagiarism means is just copying the work of an author without giving him the actual credit. They describe the first attempt to detect plagiarised segments in a text employing statistical language models and perplexity. The experiments were carried out on two specialised and 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.

Francisco *et al*. [5] 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. This has made the teachers to pay special attention on finding the plagiarism. 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. Experimental Setup

3.1. Corpus

The corpus for detecting plagiarism was collected from students of our college. Each student is assigned to write an assignment on various topics. A set of 120 students from each department were spitted into 40 groups (each group has 3 members). Department of engineering faculties like electrical, electronics, mechanical, computer science, information technology and civil with 120 strength for each. So all together we had a corpus of 120 assignments. Each group was assigned to write an assignment on different topic. They were given a week period of time and were asked to submit the assignment as a text document. It is quiet obvious that the students of all disciplines have copied their assignments. Moreover, we stick to focus on to the corpus of our faculty information technology, as we can judge the contents effectively. Only very few reasonable attempted of their own. Moreover, the assignments have stronger overlap with other groups. Table 1 shows the corpus statistics used for experimenting, illustrating the details of group ID, number of words in each assignment including the number of sentences, average number of words in each sentences.

3.2. Pre-Processing of Documents

To measure the content similarity we remove the stop words from the text document provided. This pre-processing is done in order to eliminate the relevancy among unwanted words. Stop words are ordinary or unusual words which occur in the document, which don't have significant meaning (e.g., connector words, conjunctions, single letter words). From a corpus of database, we eliminate such unwanted words [8]. We also eliminate special symbols that do not have significant part in text processing (e.g., ",", /,-, etc., in general symbols other than characters and numbers).

Table 1. Corpus used for experiments and statistics.

Corpus	Assignment Topic	Group ID	No. of Words	No. of Sentences	Average Words in Each Sentence
A	Plagiarism detection and prevention	A1	400	41	9.75
		A2	350	30	11.6
		A3	375	32	11.7
B	Comparison of 802.11	B1	500	45	11.1
		B2	470	38	12.3
		B3	425	42	10.1
C	Web Content Mining	C1	375	40	9.37
		C2	300	33	9.09
		C3	280	22	12.7
D	GSM and CDMA	D1	445	47	9.46
		D2	475	48	9.85
		D3	450	52	8.65
E	Working of CORBA and RMI	E1	600	60	10.0
		E2	624	45	13.8
		E3	580	50	11.6

Table 2. Similarity overlap among peer groups.

A1-A2					**A1-A3**					**A2-A3**				
(1)	**(2)**	**(3)**	**(4)**	**(5)**	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**
0.583	0.571	0.549	0.488	0.453	0.635	0.627	0.614	0.469	0.401	0.501	0.493	0.479	0.367	0.309
0.537	0.524	0.512	0.452	0.412	0.610	0.603	0.597	0.355	0.322	0.497	0.472	0.462	0.255	0.201
0.512	0.509	0.501	0.497	0.332	0.609	0.602	0.594	0.271	0.242	0.478	0.461	0.442	0.187	0.154

Stemming is the process of removing suffixes from words to get the common origin. In statistical analysis, it greatly helps when comparing texts to be able to identify words with a common meaning and form as being identical. Stemming identifies these common forms. We use this stemming process as a major task in plagiarism detection the documents. Stemming is useful in finding the common forms of words so as to weigh the terms effectively and to identify sentences that are similar in their root form [13].

3.3. Similarity Analysis

The system designed to detect similarity among text documents calculates content similarity among specified documents, after removal of stop words and stemming the terms. The similarity is estimated between the document samples (assignments) using various measures like cosine, dice, jaccard, hellinger and harmonic given by equations 1-5.

$$Cosine(ti,tj)=\sum_{h=1}^{k} t_{ih}t_{jh} / \sqrt{\sum_{h=1}^{k} t_{ih}^2 \sum_{h=1}^{k} t_{jh}^2} \tag{1}$$

$$Dice(ti,tj)=2\sum_{h=1}^{k} t_{ih}t_{jh} / \sum_{h=1}^{k} t_{ih}^2 \sum_{h=1}^{k} t_{jh}^2 \tag{2}$$

$$Jaccard(ti,tj)=\sum_{h=1}^{k} t_{ih}t_{jh} / \sum_{h=1}^{k} t_{ih}^2 \sum_{h=1}^{k} t_{jh}^2 - \sum_{h=1}^{k} t_{ih}t_{jh} \tag{3}$$

$$Hellinger(ti,tj)=\sqrt{\sum_{h=1}^{k} t_{ih}t_{jh}} \tag{4}$$

$$Harmonic(ti,tj)=\sum_{h=1}^{k} 2t_{ih}t_{jh} / tih+tjh \tag{5}$$

The metric calculation is done using the relevant formulas as shown above. For all the measures we take t^{ih} as the first vector (each term being normalized using the total number of words in the document) and t^{jh} corresponds to the second vector. We performed analysis by normalized method, which useful in scenarios where we might not be interested in representing very high values. We perform the similarity analysis using the formulas listed above in equations 1-5. If two documents exactly plagiarized, then we have a value of 1, 0 otherwise, from the assignments in each group, we found that there is huge overlap among them. This is shown for all 5 measures in Table 2. From the values we infer that cosine is better to reflect the similarity of content among the peer groups. Columns marked 1-5 denote the respective values measuring the similarity adopting equations 1-5. We have three groups, for example A1-A2 denotes the relevance among group A1 and A2. Similarly the relevance among other groups is named as their group ID. From the values perused from Table 2, we find that there is strong inter relevance among the groups of similar assignments. Hence it is easier to finds the best assignment among the three using cosine metric. Then next task is to measure the similarity of plagiarised documents with original source referenced.

3.4. Plagiarism Identification

To identify plagiarism online tools like article checker, duplichecker, viper, turnitin, splat available. We focus on comparing the text document with article checker and the results were shown in Table 3. To compare the source text with documents available online, commercial tools do search for the exact words or sentences online. If those words are not available, they fail report plagiarism. We made a study pertaining to assignments collected by the students and asked them even to quote the exact reference from which they have copied. All these data's were stored in the warehouse and retrieved for processing. We could find by significantly removing the stop words and applying stemming, would identify plagiarism in a better way. Combining both the parameters leads us to even further benefits.

3.5. Discussion

Now consider an example shown in Table 4, with original sentence taken from Wikepedia. This sentence is modified and fed to article checker for plagiarism detection. On close analysis we could find that both the sentences are similar in nature, however the plagiarism report differs. On close examination of original sentence, Article checker detected plagiarism of 100%. However, the same sentence when modified (but has similar meaning), plagiarised is not be detected, as words are totally recoined and hence plagiarism reported is 0%. However, when we measure the plagiarism by detecting the cosine similarity, we find they have been plagiarized to 74% (with stemming and no stop words), with stemming or no stop words 34%. Table 3 details the results carried by analyzing stop words and stemming ass parameters. From the values we found a significant change in detecting plagiarism. Hence, our work attempts solution for "copy paste" and "paraphrasing" type plagiarism.

4. Conclusions and Future Improvements

We have made an attempt to identify solutions for two different types of plagiarism attempts namely "copy paste" and "paraphrasing" type plagiarisms. For both the type the user reformulates the content in different words or styles allowing the detection tool to report negatively. We have proposed cosine metric factor to illustrate the relevance among documents. Also from the study made we found that, plagiarism is well detected through similarity analysis. The paper does not focus on plagiarism reported in other forms of content e.g., if the original content is represented in text form and the user has represented in tabular form or an images, which is left for future extensions. The paper also detects the plagiarism if only the correct source is provided. We now focus on to detect plagiarism provided if reference is valid or correct. However, improper editing of reference and detecting plagiarism from it is left for future work.

Acknowledgments

The author would like to express their sincere thanks to the anonymous reviewers for their valuable comments. The authors would also like to thank the management for providing a supporting environment.

Table 3. Plagiarism detection for corpus.

Corpus	Group ID	Article Checker	Parameter analysed				
			With Stop Words	Without Stop Words	Without Stemming	With Stemming	With Stemming & Without Stop Words
A	A1	45	43	52	47	55	60
	A2	30	28	34	32	37	36
	A3	29	26	30	30	33	35
B	B1	52	52	56	52	60	62
	B2	47	46	50	47	55	59
	B3	62	60	63	60	70	72
C	C1	38	36	38	33	42	45
	C2	27	27	29	25	33	37
	C3	32	31	29	24	34	36
D	D1	41	40	49	47	54	60
	D2	40	38	45	40	50	54
	D3	22	22	30	27	32	35
E	E1	28	28	33	30	37	42
	E2	33	30	37	35	39	39
	E3	41	40	45	42	48	49

Table 4. Original sentence and plagiarized sentence.

Original Sentence	Plagiarism, is the "use or close imitation of the language and thoughts of another author and the representation of them as one's own original work
Plagiarised Sentence	Plagiarism is imitating another authors languages and ideas and representing them as their own work
Reference Source	http://en.wikipedia.org/wiki/Plagiarism

References

[1] Abdelmalek A., Zakaria E., and Michel S., "Evaluation of Text Clustering Methods Using WordNet," *The International Arab Journal of Information Technology*, vol. 7, no. 4, pp. 349-357, 2010.

[2] Alan P. and Hamblen J., "Computer Algorithms for Plagiarism Detection," *IEEE Transactions on Education*, vol. 32, no. 2, pp. 94-99, 1989.

[3] Alberto C. and Paolo R., "Towards the Exploitation of Statistical Language Models for Plagiarism Detection with Reference," *in Proceedings of ECAI Workshop Uncovering on Plagiarism and Social Software Misuse PAN*, Greece, pp. 15-19, 2008.

[4] Allan K., Kevin A., and Bruce B., "An Automated System for Plagiarism Detection Using the Internet," *in Proceedings of World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Chesapeake, pp. 3619-3625, 2004.

[5] Francisco R., Antonio G., Santiago R., Jose L., Pedraza M., and Manuel N., "Detection of Plagiarism in Programming Assignments," *IEEE Transactions on Education*, vol. 51, no. 2, pp. 174-183, 2008.

[6] Hermann M., Frank K., and Bilal Z., "Plagiarism -A Survey," *Universal Computer Science*, vol. 12, no. 8, pp. 1050-1084, 2006.

[7] Wikipedia, available at: http://en.wikipedia.org/wiki/Plagiarism, last visited 2004.

[8] Webconfs, available at: http://www.webconfs.com/stop-words.php, last visited 2006.

[9] Jinan F., Alkhanjari Z., Mohammed S., and Alhinai R., "Designing a Portlet for Plagiarism Detections within a Campus Portal," *Journal of Science*, vol. 1, no. 1, pp. 83-88, 2005.

[10] Juan A., Nicholas C., and Rafael C., "Applying Plagiarism Detection to Engineering Education," *in Proceedings of School of Electrical and Information Engineering University of Sydney*, NSW, pp. 722-731, 2006.

[11] Nathaniel G., Maria P., and Yiu N., "Nowhere to Hide: Finding Plagiarized Documents Based on Sentence Similarity," *in Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, NSW, pp. 690-696, 2008.

[12] Ozlem U., Boris K., and Thade N., "Using Syntactic Information to Identify Plagiarism," *in Proceedings of Massachusetts Institute of Technology Computer Science and Artificial Intelligence Laboratory Cambridge*, USA, pp. 37-44, 2005.

[13] Porter F., *An Algorithm for Suffix Stripping*, Emerald Group Publishing Limited, 1980.

[14] Steve E., Vivek L., and Michelle C., "Plagiarism Detection Using Feature-Based Neural Networks," *in Proceedings of the 38th Sigcse Technical Symposium on Computer Science Education*, USA, pp. 34-38, 2007.

[15] Wadsworth, available at: http:// www.wadsworth.com/english_d/special-features/ plagiarism/, last visited 2004.

Shanmugasundaram Hariharan received his BE degree specialized in computer science and engineering from Madurai Kammaraj University, Madurai, India in 2002, ME degree specialized in the field of computer science and engineering from Anna University, Chennai, India in 2004. He holds his PhD degree in the area of Information Retrieval from Anna University, Chennai, India. He is a member of IAENG, IACSIT, ISTE, CSTA and has 8 years of experience in teaching. Currently he is presently working as associate professor in Department of Computer Science and Engineering, TRP Engineering College, India. His research interests include information retrieval, data mining, opinion mining, web mining. He has to his credit several papers in referred journals and conferences. He also serves as editorial board member and as program committee member for several international conferences and journals.