International Journal of Database Theory and Application
Vol.7, No.5 (2014), pp.161-170
http://dx.doi.org/10.14257/ijdta.2014.7.5.12

A Novel Text Copy Detection Method based on Semantic Feature

Jianjun Zhang[a,b], Xingming Sun[a,c] and Jin Wang[c]

[a]*School of Information Science and Engineering, Hunan University, Changsha 410080, China*

[b]*College of Engineering and Design, Hunan Normal University, Changsha 410081, China*

[c]*School of Computer and Software & Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing 210044, China*

Abstract

With the rapid development of the Internet, getting shared resource on the network is becoming more easy, and various plagiarize is becoming to breed, so the research of text copy detection technology is becoming more important. The traditional copy detection technology is based on term frequency statistics, and does not consider the context semantic. Some plagiarism can be easily made by replacing synonyms, changing the sentence structure, or translating from one language to another language. But the traditional copy detection technology could not detect such plagiarism. In this paper, a text copy detection method based on semantic is proposed. By using an improved TFIDF algorithm, terms could be more accurately extracted from each document in the corpus. By putting the documents corresponding to the terms one by one, a terms category is built in the database. When a document is detecting, the terms are read from the database and matched. The testing results show that, compared to the traditional TFIDF algorithm, the improved method could more accurately detect the plagiarism.

Keywords: *plagiarism, feature extraction, TFIDF algorithm, copy detection*

1. Introduction

In recent years, with the rapid development of the Internet and increasingly rich Web resources, a large amount of information appear in many forms on the Internet. These forms are text, image, audio, video and so on. Because text is easily accessible, informative and timely, so text is one of network information copied more easily.

On one hand, because these text messages share on the Internet, people can very easily find the needed information by doing a keyword search on the Internet. But on the other hand, due to the open of the Internet, some people could steal other's work as their own by simply illegal copying or plagiarizing. There are some main plagiarism types. One is almost unaltered copying the original text; the other plagiarism types are changing some sentences' order, replacing some words with synonyms or translating text from one language to another. But no matter what kind of plagiarizing, plagiarism brings to the information owners not just the loss of benefit, more spiritual harm.

The purpose of this paper is to propose an effective text copy detection technology based on semantic features words, and then apply it to digital libraries, Internet information retrieval, and online articles submission system. The results of this research

ISSN: 2005-4270 IJDTA

can be used to detect whether a paper is a plagiarism work, and as an evidence for legally recognized plagiarism.

The rest of paper is organized as follows. We discuss the related work on searchable encryption in Section 2. In Section 3, we introduce the system model, threat model and our design goals and then briefly describe some notations and background knowledge used in our paper. Section 4 shows the detailed construction of semantic keyword-based search scheme. Section 5 presents performance analysis and Section 6 concludes the paper finally.

2. Related Work

Copy detection, also known as plagiarism detection or Duplicate Detection, is used to detect whether a document is a plagiarism [1]. In this paper, we discuss text copy detection for Chinese natural language.

Research on natural language text copy detection began in the 90 's of the last century. In 1993, Manber developed a program called "Sif" (later renamed "Siff") [2]. This tool is primarily used to find similar content in the vast corpus of documents. The concept of text copy detection was not explicitly proposed, but the Sif tool raised "approximate fingerprints" concept, whose principle was to measure the similarity of documents by using string matching method. In 1995, Brin and others firstly proposed the concept of copy detection and developed the COPS(Copy Detection System) in the research of "digital library" at Stanford University [3]. At the same year, Garcia-Molina and Shivakumar proposed the SCAM (Stanford Copy Analysis Method)model [4,5]. Experiments showed that the SCAM's performance was better than the COPS system. Later, they proposed the DSCAM (Distributed Stanford Copy Analysis Method) model based on the SCAM model [6], and extended the detection range from the single registration database to the distributed database, as well as text copy detection on the Web. In 1996, by using digital fingerprinting methods, Heintze developed the KOALA plagiarism recognition prototype system [7], and released the system on the Internet for free testing. At the same year, Wise developed YAP1, YAP2, YAP3 series tools [8]. In 2000, Monostori and Zaslavsky, by using the suffix tree, developed the MDR (Match Detect Reveal) system [9,10], which could be used for document overlapping identification with high accuracy. In 2001, Finkel [11] proposed SE (Signature Extraction) method. In 2002, Chowdhury and others developed the I-Match system, based on digital fingerprinting technology [12], to implement the fast detection algorithm of duplicate documents in a large scale documents collection. At the same year, Hoad and Zobel [13], by using digital fingerprints combined with the word frequency statistic methods, solved the problem of co-derivatives identification. In 2003, Schleimer and others put forward Winnowing algorithm [14], based on digital fingerprints, to improve the document copy detection accuracy. In 2007, Gurmeet Singh Manku and others proposed Smihash method [15] to detect the almost duplicate pages. Since 2007, the world's most authoritative English detection system, Turnitin, providing service of originality checking and plagiarism prevention, has been applied to more than 50 countries and regions scientific research institutions to detect text copy or plagiarism. In 2008, the iParadigms Corporation developed a tool, CrossCheck [16], which is designed to help the academic publishers verify the originality of published documents. CrossCheck consists of two parts: a Web-based testing tool and a huge database of the global academic publications [17].

3. Priori Knowledge

3.1. Chinese Word Segmentation

Chinese word segmentation refers to segmenting a sequence of Chinese characters into single Chinese words. As well known, in English, words are separated by spaces as natural demarcation breaks. In Chinese, however, character, sentence and paragraph can be simply delimited by obvious demarcations breaks, while only words haven't demarcation breaks. In English, there is also a problem of phrases dividing. But the Chinese Word Segmentation is more complex and difficult than it.

In this paper, the corpus is the SOUGOU Chinese experimental corpus, in which the documents must be processed with word segmentation. The word segmentation system was designed based on ICTCLAS developed by the Chinese Academy of Sciences. This word segmentation system can be used to process batch documents in the corpus.

3.2. Semantic Feature Extraction

Semantic feature extraction is the core of text copy detection technology. Its function is to preprocess the documents, by using reasonable and efficient algorithms, and extract semantic keywords representing the entire document. There are many ways to extract semantic features: TFIDF algorithm, key frequency method, document frequency, mutual information method, expected cross entropy method, information gain method, X2 statistic method, and so on.

3.3. Retrieval Algorithm

Semantic retrieval is a retrieval technology based on the concept and the correlation, and analyses the information objects and the retrieval requests from the perspective of semantic understanding. It is very important to select an appropriate retrieval algorithm. For example, you can create an index based on Tree structure for the extracted keywords. So you can quickly and easily find the corresponding text by searching the keywords. Retrieval algorithm determines the speed and the efficiency of queries, so you should select an efficient retrieval algorithm.

3.4. Fuzzy Matching and Exact Matching

Fuzzy matching does not require an exact matching, the purpose of which is to get matching objects with a certain similarity. Fuzzy matching can be used to query synonyms. Even if the plagiarists replace some words with their synonyms, but the meaning is similar, so this plagiarism can be detected by using fuzzy matching. Exact matching is strict and accurate, and matching can be achieved only when the available words and the query words are same. Although the exact matching has the higher accuracy, the fuzzy matching has a wide application range. So they have advantages and disadvantages.

4. Design of detection system

4.1. The TFIDF Algorithm

TFIDF algorithm is by far the most efficient method for calculating the weight of the word. TF (Term Frequency) is the frequency of the term t appearing in the document d.

IDF (Inverse Document Frequency) is the number of documents containing the term t. IDF can be used to calculate the feature term's ability to distinguish document categories. IDF=log (N/n), wherein, N is the number of documents, and n is the number of documents that contain the term t. If the frequency of a term appearing in a particular document, TF, is higher, and it is rarely seen in other documents, the term has good ability to distinguish categories. So the term is the keyword of the document, and suitable for text classification.

4.2. The Improved TFIDF Algorithm

Traditional TFIDF approach has certain defects, mainly reflected in the inverse document frequency-IDF. IDF is the number of documents containing the feature word. According to the traditional TFIDF algorithm, if a term t appears many times in a document, it's IDF is smaller, so its weight is smaller. However, this is obviously incorrect because the calculating of IDF does not consider the distribution of the feature term in a category and between categories. If a feature term appears many times in a certain category while rarely appearing in other categories, its classification ability is very strong, and should be given a higher weight. So the accuracy of the traditional TFIDF algorithm is not high.

In this paper, we proposed a new TFIDF method, C-TFIDF, in which the weight of category is added for each feature term. The term frequency is called "C-TF", and the inverse document frequency is called "C-IDF". Let the number of documents is N, and the number of documents that contains the feature term t is n. In a category C, if the number of documents that contains the feature term t is m, so the inverse document frequency in category C is: C-IDF= log[(m/n)*N]. If there is a category C, in which the number of documents containing the term t is larger, while in other categories the number is smaller, so the term t has the good ability to distinguish categories and could represent the text feature of category C. In other categories, if the number of documents containing the term t is k, the variant of the formula is: C-IDF=log[(m/m+k)*N], in which m+k=n. The experimental results showed that the improved TFIDF algorithm, C-TFIDF, has a better ability to extract feature.

4.3. System Design

The flow chart of the text copy detection system is shown in Figure 1. Firstly, we built a text corpus, and then preprocessed the corpus. the preprocess included two steps: first, Chinese word segmentation for all documents in the corpus; second, feature extraction for the corpus by using CTFIDF. Then we created a the feature term library in the database. After the interface of the text copy detection was designed, the documents could be detected and detection reports would be generated.

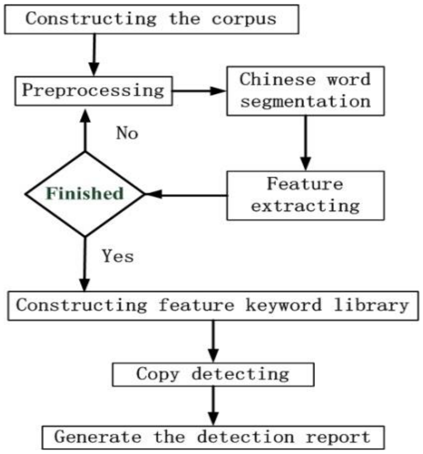
Constructing the corpus
Preprocessing
Chinese word segmentation
No
Finished
Feature extracting
Yes
Constructing feature keyword library
Copy detecting
Generate the detection report

Figure 1. The Flow Chart of Detection System

4.4. The Proposed Detecting Solution

The proposed detecting solution includes four components: building of the corpus, preprocessing of the corpus, constructing of the feature term library and the implement of the copy detection algorithm.

4.4.1 Constructing the Corpus

We selected eight categories of documents from the SOUGOU Chinese corpus. Each category has 1000 articles. The categories are mainly financial, health, education, sports, military, tourism, culture and recruiting.

4.4.2 The Preprocessing of the "Corpus"

The preprocessing of the corpus includes two steps: Chinese word segmentation and feature extraction.

- Chinese word segmentation

The Chinese word segmentation was designed based on the ICTCLAS system developed by the Chinese Academy of Science. By calling the ICTCLAS dynamic link library, we could make Chinese word segmentation for a single document with C# language. On this basis, we developed a system which could make word segmentation for each text file in a folder.

- Feature extraction

Feature extraction method used in this paper is the improved TFIDF method , in which the formula of IDF was modified to accurately set a weight for each word. So the feature extraction of each document could be finished.

4.4.3. Building the feature word library

After implementing the improved TFIDF algorithm, we built a feature word library in the database in order to save all each document and their feature words. We used MySQL as the database storing the documents and its feature words. By establishing the relationship between the documents and its feature words one by one, we could easily locate the position of copy when we make document copy detecting.

4.4.3. Text Copy Detection

Before detecting, the designed system will select the highest weight 5 keywords words in a detected document, then search the feature words of each document in the database with exact matching, and finally display the detecting results. The results include the number of documents containing each feature word and the number of documents containing the same feature words. In addition, the system will also display the number of the documents containing each keywords.

In the designed system, the plagiarism threshold is 3. If the number of the same feature words between the detecting document and a document in the corpus is equal or larger than 3, the system will display "this document may be a copy of some documents", and return the location of documents in the corpus. Then we navigate to the document in the corpus to find how many similarities between the document and the detecting document, thus finishing the detecting. If there is no document containing the 3 or less feature words of the detecting document, then it can be considered to be no plagiarism in the detecting document.

Before detecting, we must firstly judge the detecting document's category, and then detect the document in the appropriate category. If we don't select the appropriate document category, the detection results could be incorrect because the extracting feature words are incorrect when the feature words extracting is made in the inappropriate category. So before detecting, we could judge the document's category from its title or the abstract. Only by selecting the appropriate detecting document category, we could get more accurate results.

5. Performance Analysis

The testing was divided into two parts. The first part was to compare the accuracy of the feature words extracting with the traditional TFIDF algorithm and the improved TFIDF algorithm, and compare the keywords extracted with the two methods and artificial extracting. The second part was to test how much the number of keywords has impact on the accuracy of the text copy detection system. We used 120 testing documents selecting from 8 categories corpus, 15 documents from each category, to ensure the completeness and accuracy of the testing results.

5.1. Stemming Process

In order to compare the accuracy of the feature words extracting with the traditional TFIDF algorithm and the improved TFIDF algorithm, there must be a standard. We selected the artificial extracting keywords as the comparing standard. If the number of artificial extracting keywords is N and the number of same keywords extracted with the traditional TFIDF algorithm is p ($p \leqslant N$), the accuracy rate of the traditional TFIDF algorithm is (p / N)*100%. Similarly, if the number of same keywords extracted with the improved TFIDF algorithm is q ($q \leqslant N$), the accuracy rate is (q / N) *100%. We set the number of extracting keywords 10, and the testing results is shown in Table 1 (the testing results of the traditional TFIDF algorithm) and Table 2 (the testing results of the improved TFIDF algorithm).

In Table 1, the first line is the number of keywords extracted with the traditional TFIDF algorithm. The second line is the number of documents containing the extracted keywords shown in the first line. The sum of the testing documents is 120. For each document, because the number of artificial extracting keywords is 10, if the number of the extracted keywords is x, so the accuracy rate is (x/10)* 100%. The accuracy rate is shown in third line. The average accuracy rate is calculated by [(1x0%) + (5x10%) +...+ (3x90%) + (1x100%)] ÷ 120=44.92%, shown in the fourth line.

Table 1. The Accuracy Rate of Keywords Extracted with the Traditional TFIDF Algorithm

The number of correct keyword	0	1	2	3	4	5	6	7	8	9	10
The number of documents	1	5	20	12	24	21	17	10	6	3	1
The accuracy rate	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
The average accuracy rate	44.92%										

The accuracy rate of keywords extracted with the improved TFIDF algorithm is shown in Table 2. From this table, we can find the accuracy rate is 59.83%, higher than the accuracy rate of keywords extracted with the traditional TFIDF algorithm.

Table 2. The Accuracy Rate of Keywords Extracted with the Improved TFIDF Algorithm

The number of correct keyword	0	1	2	3	4	5	6	7	8	9	10
The number of documents	0	2	5	10	15	12	20	26	18	10	2
The accuracy rate	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
The average accuracy rate	59..83%										

5.2. The Impact of the Keyword Number on the Detection Accuracy

The number of keywords is a variable having much impact on the efficiency of the detecting system and the accuracy rate of keywords extracted. If the number of keywords is less, for example, a document corresponding to a keyword , the keyword must be the highest weight keyword in a document. When detecting a document, we will think it may be a copy of a document if the corpus contains a document containing the same keyword. Otherwise, there is no plagiarism. In fact, however, this method is not accurate because it does not consider the impact of other high weight keywords in a document. For two documents, we could not conclude that there is a plagiarism only because they have the same the highest weight keyword. Similarly, we could not conclude that there is no plagiarism only because the two document have not the same highest keyword, because they may have many same keywords. Moreover, if the number of the keyword is less, the detecting threshold is not easily set, and someone can easily evade detection by changing the keywords.

Similarly, more keywords may lead to some problems. If the number of the keywords is 20, that is, each document corresponds to 20 keywords, we need to extract 20 keywords from the detecting document. As well known, the number of the keywords in a document is normally less than 10. If we extract 20 keywords form a document, there are 10 higher keywords and 10 irrelevant keywords. If the top higher weight 10 keywords of a document and the low higher weight 10 keywords of another document are same, the detecting result will be 50% similar. However, this result is clearly error.

Thus, selecting the number of keywords is very important. In General, the best number of artificial extracting keywords is 3~5, like the number of keywords we wrote in a document. The best number of the machine extracting keywords is 5~10. We used 3, 5, 8, 10, 15 as the number of keywords to detect 120 documents. For comparison, the plagiarism threshold is set to 60%, meaning that there exists plagiarism if two document have 60% same keywords. The detecting result is shown in Table 3.

Table 3. The Number of Plagiarism Documents Detected with Different Number Keywords

The number of keywords	3	5	8	10	15
The number of plagiarism document detected	33	23	19	14	6

(1) When the number of keywords is set to 3, there exists plagiarism if two documents have 2 (3×60%=1.8) same keywords. From the Table 3, in 120 documents, we found that there were 33 documents existing plagiarism.

(2) When the number of keywords is set to 5, there exists plagiarism if two documents have 3 (5×60%=3) same keywords. From the Table 3, in 120 documents, we found that there were 25 documents existing plagiarism.

(3) When the number of keywords is set to 8, there exists plagiarism if two documents have 5 (8×60%=4.8) same keywords. From the Table 3, in 120 documents, we found that there were 19 documents existing plagiarism.

(4) When the number of keywords is set to 10, there exists plagiarism if two documents have 6 (10×60%=6) same keywords. From the Table 3, in 120 documents, we found that there were 12 documents existing plagiarism.

(5) When the number of keywords is set to 15, there exists plagiarism if two documents have 9 (15×60%=9) same keywords. From the Table 3, in 120 documents, we found that there were 6 documents existing plagiarism.

When the number of keywords is set to 3, 33 plagiarism documents were detected, meant that some documents were mistakenly detected as plagiarism. When the number of keywords is set to 15, 6 plagiarism documents were detected, meant that some documents were not detected as plagiarism. When the number of keywords is set to 5~10, 19 plagiarism documents were detected. The result was close to the actual number of plagiarism documents.

If the number of detected plagiarism documents is p, and the actual number of plagiarism documents is x, we used 'p/x' to measure the correct detecting rate of replication. The testing result is shown in Table 4.

Table 4. The Detection Accuracy Rate

The number of keywords	3	5	8	10	15
The number of correctly detected documents	8	12	15	9	5
The accuracy rate	40%	60%	75%	45%	25%

As table 4 showing, when the number of keywords is set to 8, the correct detecting rate is the highest, reaching 75%. When the number is set to 15, the rate is the lowest, merely 25%. When the number of keywords is set to 5~10, the correct detecting rate reaches 40% to 60%,which is an acceptable accuracy. Thus, it can be concluded that when the number of keywords is set to 5~10, the correct detecting rate is proper, suitable for the number of extracted feature words.

Therefore, the testing result shows not only the advantage of the improved TFIDF algorithm, but also the keyword number impact on the correct detecting when the plagiarism threshold is set to a value. In actual detecting, it is very important to choose the number of keywords and the plagiarism threshold. Only when the number of keywords and the threshold are properly set, we will get the best detecting result.

6. Conclusion

In this paper, we proposed an improved TFIDF algorithm, which can be used to accurately extract the feature words from the documents in the corpus. The algorithm can be used to detect text plagiarism. The testing result shows not only the advantage of the improved TFIDF algorithm, but also the keyword number impact on the correct detecting when the plagiarism threshold is set to a certain value.

However, there are some problems in this method. The principle of the detecting system is based on keyword matching, by which we calculated the similarity between two documents and judged whether there exists plagiarism. Although this approach is feasible, but we did not consider the context of the sentence structure and semantics, which is the focus of our future work.

Acknowledgements

This paper is a revised and expanded version of a paper entitled “Semantic Keyword-based Text Copy Detection Method” presented at SoftTech 2014, Yeosu, Korea, May 8-10, 2014. This work is supported by the NSFC (61232016, 61173141, 61173142, 61173136, 61103215, 61373132, 61373133, 61402234), GYHY201206033, 201301030, 2013DFG12860, BC2013012, PAPD fund, Hunan province science and technology plan project fund (2012GK3120), the Scientific Research Fund of Hunan Provincial Education Department (10C0944), the Prospective Research Project on Future Networks of Jiangsu Future Networks Innovation Institute (BY2013095-4-10) and China Meteorological Administration under CMA grants (2014 MC 16, M43).

References

[1] J. P. Bao, J. Y. Shen, X. D. Liu and Q. B. Song, “A Survey on Natural Language Text Copy Detection”, Journal of software, vol. 14, no. 10, **(2003)**, pp. 1753-1760.

[2] U. Manber, “Finding similar files in a large file system”, In: Proceedings of the Winter USENIX Conference, **(1994)**, pp. 1-10.

[3] S. Brin, J. Davis and H. G. Molina, “Copy detection mechanisms for digital documents”, In: Proceedings of the ACM SIGMOD Annual Conference, **(1995)**, pp. 12-17.

[4] N. Shivakumar and H. G. Molina, "SCAM", A copy detection mechanism for digital documents. In: Proceedings of the 2nd International Conference in Theory and Practice of Digital Libraries, **(1995)**, pp. 22-28.

[5] N. Shivakumar and H. G. Molina, "Building a scalable and accurate copy detection mechanism", In: Proceedings of the 1st ACM Conference on Digital Libraries, **(1996)**, pp.10-22.

[6] H. G. Molina, L. Gravano and N. Shivakumar, SCAM: Finding document copies across multiple databases", In: Proceedings of the 4th International Conference on Parallel and Distributed Systems, **(1996)**, pp.122-125.

[7] N. Heintze, "Scalable document fingerprinting", In: Proceedings of the 2nd USENIX Workshop on Electronic Commerce, **(1996)**, pp. 1-5.

[8] M. Wise and J. YAP3, "Improved detection of similarities in computer programs and other texts", In: Proceedings of the SIGCSE, **(1996)**, pp.130-134.

[9] K. Monostori, A. Zaslavsky and H. Schmidt, "Match Detect Reveal: Finding overlapping and similar digital documents", In: Proceedings of the Information Resources Management Association International Conference, **(2000)**, pp. 88-92.

[10] K. Monostori, A. Zaslavsky and H. Schmidt, "Parallel overlap and similarity detection in semi-structured document collections", In: Proceedings of the 6th Annual Australasian Conference on Parallel and Real-Time Systems, **(1999)**, pp. 65-71.

[11] R. A. Finkel and A. Zaslavsky, "Signature extraction for overlap detection in documents", Twenty-fifth Australasian Computer Science Conference, **(2002),** Melbourne, Australian, pp. 59-64.

[12] A. Chowdhury and F. O. Grossmand, "Collection statistics for fast duplicate document detection", ACM Trans. Inf. Syst., vol. 20, no. 2, **(2002)**, pp.171-191.

[13]T. C. Hoad and J. Zobel, "Methods for identifying versioned and plagiarism documents", Journal of the Amer. Soc. for Inf. Sci. and Technol., vol. 54, no. 3, **(2002)**, pp.203-215.

[14] S. Schleimer and D. S. Wilkerson, "Winnowing: local algorithms for document fingerprinting", ACM SIGMOD 2003, **(2003)**, pp.204-212.

[15] G. S. Manku, A. Jain and A. Das Sarma, "Detecting Near-Duplicates for Web Crawling", The 16th International World Wide Web Conference, **(2007)**; Banff, Alberta, CANADA.

[16] M. Qin, "Research on document copy detection technology and application in academic supervision", Zhengzhou, ZhengZhou University, **(2012)**.

[17] T. Li, "Research and implement of the copy detection system for acacemic papers based on semantic structure", Beijing, Beijing University of Posts and Telecommunications, **(2009)**.

Authors

Jianjun Zhang received his BS in applied mathematics from Xinyang Normal University, China, in 1997; his MS in computer science and technology form Yunnan Normal University, China, in 2000. He is currently pursuing his MS in computer science and technology at the School of Information Science & Engineering, Hunan University, China. His research interests includes network and information security, software engineering.

Xingming Sun received his BS in mathematics from Hunan Normal University, China, in 1984; his MS in computing science from Dalian University of Science and Technology, China, in 1988; and his PhD incomputing science from Fudan University, China, in 2001. He is currently a professor at the College of Computer and Software, Nanjing University of Information Science and Technology, China. In 2006, he visited the University College London, UK; he was a visiting professor in University of Warwick, UK, between 2008 and 2010. His research interests include network and information security, database security, and natural language processing.