Copy detection in Chinese documents using the Ferret

JunPeng Bao (baojp@mail.xjtu.edu.cn)
Department of Computer Science & Technology, Xi'an Jiaotong University, Xi'an 710049, China

Caroline Lyon (c.m.lyon@herts.ac.uk) and
Peter C. R. Lane (peter.lane@bcs.org.uk)
School of Computer Science, University of Hertfordshire, Hatfield AL10 9AB, UK

Abstract. The Ferret copy detector has been used for some years on English texts to find plagiarism in large collections of students' coursework. This article reports on extending its application to Chinese. Corpora of coursework from two Chinese universities have been collected, and our experiments show that the Ferret can find both artificially constructed plagiarism and also actually occurring, previously undetected plagiarism. We discuss issues of representation, focus on the effectiveness of a sub-symbolic approach, and show that the Ferret does not need to find word boundaries.

1. Introduction

Detecting the presence of copied material in documents is a problem confronting many disciplines. In education students may plagiarise, as may writers in academic journals (Giles, 2006). In the commercial world, copying is found in theft of copyright or intellectual property. Detecting copied, or duplicated, material is also of importance in managing language resources, to locate and highlight links between related documents.

Ferret (Lyon et al., 2001; Lyon et al., 2006) is a tool for detecting similar passages of text in large collections of documents. It has been used successfully on English texts for some years. It is a free, stand alone system designed to be run by naive users on their own PCs, giving immediate results (Lane et al., 2006). It enables large numbers of documents, such as essays from a large cohort of students, to be analysed quickly, and can also be used to identify plagiarism in programming code. This article reports that an adapted version of Ferret performs effectively on Chinese texts. Corpora of students' coursework from two Chinese universities have been collected, and we applied Ferret to investigate the detection of plagiarism. Our experiments show that Ferret can find both artificially constructed plagiarism as well as actually occurring, previously undetected plagiarism.

Another well known system for copy detection is Turnitin (2006), which uses an enormous database of material off the web and previous

student work, against which it compares current student work. However, documents have to be submitted to Turnitin for processing, and there is a commercial charge. A comparison of Ferret, Turnitin and other systems is given in (Lyon et al., 2003). Alternative approaches look at semantic similarities between pairs of documents (Bao et al., 2006; Bao et al., 2004a). Copy detection in code is reported on in (Malphol, 2006).

We are not aware of any other system for detecting copied material in Chinese.

2. Outline of the Ferret system

The Ferret copy detector takes a set of files and computes a measure of similarity for each pair. The first stage in the process is to convert each document to a set of overlapping trigrams. Thus, a sentence like:

A storm is forecast for the morning.

will be converted to the set of trigrams:

```
a storm is      storm is forecast     is forecast for
forecast for the        for the morning
```

Then the set of trigrams for each document is compared with all the others, and a measure of resemblance for each pair of documents is computed. Usually, the results are presented in a ranked table with the most similar pairs at the top. Any pair of documents can be displayed and compared side by side with matching passages highlighted. Screen shots can be seen at `http://homepages.feis.herts.ac.uk/~pdgroup`. If two documents are written independently there will be a sprinkling of matching trigrams, but if there has been collusion or copying there will be solid passages that are all or mostly highlighted indicating a quantity of matching word sequences. The similarity measure still records a significant value even if some words are replaced.

We use a measure of similarity the *Resemblance metric* (Broder, 1998), also known as the Jaccard coefficient (Manning and Schütze, 2001, page 299). Informally, the measure compares the number of matches between the elements of two sets of trigrams, scaled by joint set size.

Let S(A) and S(B) be the set of trigrams from documents A and B respectively. R(A,B), the resemblance between A and B, is defined as

$$R = \frac{|S(A) \cap S(B)|}{|S(A) \cup S(B)|} \tag{1}$$

$0 \leq R \leq 1$ Two identical documents have an R-score of 1.

3. Adapting Ferret for Chinese documents

We can adapt Ferret to work on different kinds of text by basing the definition of a trigram on different kinds of *token*. Such an approach has already been used for computer programs (Lane et al., 2006).

As is well known, Chinese words may consist of one, two or up to four characters, with no white space or other marker between words. However, Chinese and English share a crucial characteristic: both are sequences of discrete data. In English the data items, the tokens, are words, while in Chinese they can be characters. A text in either language can be taken as a sequence of tokens. We can then apply the same principle to detect copied material; as we shall show it is not necessary to find word boundaries, during processing.

Three strategies to process the strings of characters that make up a Chinese sentence can be seen as follows:

Naive strategy: Chinese characters are treated like English characters; sequences are segmented by taking as a token boundary any element that is not a Chinese character: white space, punctuation, numbers etc.

Single character strategy: Instead of finding words, characters are processed singly. Each individual character in the text file is treated as a token.

Dictionary strategy: Based on a Chinese dictionary, a sentence is separated into a sequence of words, identified in the dictionary. A report on advances in word segmentation is described by Gao et al. (2006). We do not use this strategy here, as the simpler methods listed above are effective.

To contrast these approaches see Figure 1(a) which shows a Chinese sentence. In English this means "TSP is an NP problem (TSP means the Travelling Salesman Problem)". Using the naive strategy, we get three Chinese tokens in the sentence because it is segmented by two punctuation marks, as shown in Figure 1(b). With the single character strategy, we get 8 Chinese tokens because there are 8 Chinese characters in it, as shown in Figure 1(c). With the dictionary strategy, we get 5 Chinese words as tokens, as shown in Figure 1(d).

The same core algorithm can be used for detecting similar passages, using different types of tokens, as described below.

typeText A token is a sequence of items with boundaries marked by white space or punctuation marks. In English a token is a sequence of alphabetic characters constituting a word. In Chinese a token is a sequence of characters demarcated in the same way (the naive strategy). This is illustrated in Figure 1(b). We refer to the Ferret system using this type as Ferret_T.

(a)

TSP问题，即Traveling Salesman Problem，是一个NP问题。

(b)

"TSP问题" "即Traveling" "Salesman" "Problem" "是一个NP问题"

(c)

"问" "题" "即" "是" "一" "个" "问" "题"

(d)

"问题" "即" "是" "一个" "问题"

(e)

"TSP" "问" "题" "即" "Traveling" "Salesman" "Problem" "是" "一" "个"
"NP" "问" "题"

Figure 1. A Chinese sentence (a) with its words parsed with different strategies: (b) using naive strategy, (c) using single-character strategy, (d) using dictionary strategy, and (e) using mixed strategy.

typeChinese A token is a single Chinese character without any other symbols. Chinese characters are processed singly and any alphabetic characters are ignored. This is illustrated in Figure 1(c). We refer to the Ferret system using this type as Ferret_C.

typeMix A token is either a sequence of consecutive alphabetic characters (an English word), or a single Chinese character. This type of mixed text with a few foreign terms is commonly found in modern Chinese documents, especially in scientific literature. This is illustrated in Figure 1(e). We refer to the Ferret system using this type as Ferret_M.

In the case of typeMix, Ferret combines the naive strategy and single character strategy so that it processes English text with the naive strategy and Chinese with the single character strategy. That enables Ferret to avoid missing out English words in a Chinese document. For example, Figure 1(a) is a Chinese sentence including English words. Figure 1(c) shows that treating the sentence as typeChinese loses some words, and may lead to potential errors.

Table I. Details of a sample of the corpora

Corpus	Total files	Number of tokens* Average	Max	Min	Pseudo-plagiarism	Plagiarised document pairs
Xi04	320	4136	25474	104	No	N/A
Gu05	124	1125	21762	102	No	N/A
Xi04_P50	156	4600	13756	191	Yes	1031
Xi04_P500	156	5801	13756	1448	Yes	1188

* A token is a single Chinese character or an English word.

4. Experiments

We have run experiments on two raw Chinese corpora, collected in 2004 and 2005 from two Chinese universities. Full details are given in a technical report (Bao et al., 2006). Xi04 is a collection of 320 individual reports on artificial intelligence topics. Gu05 is a collection of 124 reports on solving mathematical questions.

In both cases the raw materials are MSWord files. The first stage in processing with Ferret is to convert these .doc files to .txt. We use Antiword `http://www.win_eld.demon.nl/` to convert them into plain texts in UTF-8 encoding.

Pseudo-plagiarised texts were created by taking parts of documents and copying and pasting them into other documents. Hence, we get a corpus including pseudo-plagiarised documents named as Xi04_Pn, where n indicates the minimum size of each copied unit in characters.

Our first experiment explores the effect of the different strategies and document types for processing unsegmented strings of Chinese characters, using Ferret_T, Ferret_C, and Ferret_M.

We processed the complete set of documents for the two corpora, with the three forms of Ferret, and recorded the number of times the Resemblance metric for a pair of documents falls within a range $[a, b)$, where a and b are numbers between 0 and 1, and a number r falls within the range $[a, b)$ if $a \leq r < b$ (Table II). We initially checked samples manually and found that results matched our subjective judgements.

The score distribution of Ferret_T differs from that of Ferret_C and Ferret_M. As expected, the rank of Ferret_C is similar to that of Ferret_M, since the documents are mainly composed of Chinese characters mixed with just a few English words. The documents in Gu05 gave comparable results (Bao et al., 2006).

As well as the artificially constructed plagiarised texts, we also found copied sections in the students' reports that had not been noticed previously.

Table II. The Ferret resemblance scores distribution on Xi04

Score	Ferret_T		Ferret_C		Ferret_M	
interval	count	proportion	count	proportion	count	proportion
$[0, 0.01)$	49910	0.977861	15205	0.297904	15627	0.306172
$[0.01, 0.02)$	382	0.007484	12503	0.244965	13316	0.260893
$[0.02, 0.04)$	351	0.006877	18628	0.364969	17692	0.34663
$[0.04, 0.06)$	150	0.002939	2451	0.048021	2253	0.044142
$[0.06, 0.08)$	70	0.001371	741	0.014518	676	0.013245
$[0.08, 0.1)$	55	0.001078	396	0.007759	399	0.007817
$[0.1, 0.3)$	79	0.001548	1010	0.019788	972	0.019044
$[0.3, 1.0]$	43	0.000844	106	0.002076	105	0.002057

Table III. The maximum F1 values for corpora with different amounts of copied material. (F1 is the F1 score, P precision, R recall, and θ the threshold.)

Corpus	Ferret_T				Ferret_C			
	F1	P	R	θ	F1	P	R	θ
Xi04_P50	0.59	0.98	0.42	0.01	0.30	0.66	0.20	0.05
Xi04_P100	0.85	0.97	0.76	0.01	0.51	0.53	0.49	0.04
Xi04_P300	0.97	0.95	0.99	0.01	0.83	0.87	0.80	0.05
Xi04_P500	0.98	0.99	0.97	0.02	0.92	0.91	0.92	0.05

4.1. Optimum thresholds

We can be sure that two documents are very similar when the Ferret score is high. But in practice many plagiarised documents copy part of their contents from others, not the whole paper, so that their scores are in a mid range, and Ferret needs a lower threshold to detect them. The optimum threshold for Ferret has to be fixed empirically. In the second set of experiments, we find an appropriate threshold for our Chinese corpora.

The series of artificially constructed corpora Xi04_Pn are used here to determine parameters of Ferret. We have not taken into our calculations the naturally occurring plagiarism.

We compute three measures to determine the performance of Ferret: precision (P), recall (R) and F1. Precision is the proportion of plagiarised pairs detected by Ferret which are correctly identified. Recall is

Table IV. Plagiarism detection for different thresholds on Stu04Rpt_P500. (F1 is the F1 score, P precision, and R recall.)

Threshold	Ferret_T			Ferret_C			Ferret_M		
θ	P	R	F1	P	R	F1	P	R	F1
0.01	0.96	0.99	0.98	0.10	1.00	0.18	0.10	1.00	0.19
0.02	**0.99**	**0.97**	**0.98**	0.14	1.00	0.25	0.15	1.00	0.26
0.03	1.00	0.87	0.93	0.33	0.98	0.49	0.37	0.99	0.53
0.04	1.00	0.72	0.84	0.65	0.97	0.78	0.69	0.973	0.81
0.05	1.00	0.62	0.76	**0.89**	**0.92**	**0.90**	**0.91**	**0.92**	**0.92**
0.06	1.00	0.55	0.71	0.98	0.83	0.90	0.99	0.83	0.90
0.07	1.00	0.49	0.66	1.00	0.72	0.84	1.00	0.72	0.83
0.08	1.00	0.44	0.61	1.00	0.63	0.77	1.00	0.62	0.76
0.09	1.00	0.38	0.55	1.00	0.54	0.70	1.00	0.53	0.69

the proportion of the plagiarised pairs which Ferret detects. F1 is a standard metric which takes into account both precision and recall, which may have opposing tendencies.

We interpret the results from Ferret by setting a threshold θ, so that any pair of documents whose resemblance score exceeds that threshold is suspected of containing copied material. The optimum value for the threshold is that which leads to the greatest F1 value.

Table III shows the greatest F1 value of Ferret on Xi04. Table IV shows the trends of Ferret precision, recall, and F1 for different thresholds on Xi04_P500, which are very similar to the trends on other corpora (Bao et al., 2006)

The F1 value of Ferret_T reaches a maximum around 0.01 to 0.02 as shown in Table III. Ferret_C and Ferret_M reach a peak around 0.04 to 0.05. Ferret can find copied material with both high precision and recall at or above those thresholds.

We see that the F1 score for Ferret_T is higher than the others, particularly for smaller amounts of copied text. With the shorter tokens used in Ferret_C and Ferret_M there will be some naturally occurring matches in non-copied text, whereas there is much less likely to be a match with the longer token in Ferret_T, so the threshold can be lower. This suggests that the longer segments using the naive strategy may be the most useful, but in practice it may not be the case. When there is an attempt to deceive there may be a number of minor changes that undermine the use of the longer token, as discussed later.

4.2. Investigating thresholds

The Ferret optimum threshold is found to be consistent across different sized document sets. This shows that customised thresholds can be set by analysing a small sample of a large set of documents.

We try to find the lower limit for detecting copied passages in Chinese. When the number of copied tokens is between 300 and 500, Ferret_T is still able to find most of them, but Ferret_C and Ferret_M fail to find nearly half of them. When the number is less than 300, it is hard for Ferret to find most of them. It seems that 500 tokens is the lower limit for Ferret_C and Ferret_M on these data at the optimum threshold around 0.05, which account for about 10% tokens of a document (i.e. 5% of a document pair) in our corpora. Ferret_T has a lower limit at the optimum threshold around 0.01. This contrasts with the level at which copying is detected in English, which is typically about 3-4% of words (Lyon et al., 2003, Section 5.3), in documents 10,000 words long. Thus Ferret can detect plagiarised documents with a high probability as long as the size of the copied content in them is greater than the lower limit.

We checked all of the document pairs that contain more than 1000 copied tokens but fail to be detected by Ferret, and found that they are all related to 4 documents which contain large segments of C-style source code in them. Ferret_C ignores any non-Chinese character so that it cannot detect the copied code in the plagiarised documents.

Since Ferret_M considers each Chinese character as a token the size of a document's tuple set is much larger than that of Ferret_T. If the copied section consists mainly of code, then Ferret_M gets a small R-score, which causes its failure. However, the smaller size of the tuple set does not produce such a low R-score for Ferret_T so it detects the copied code, and seldom misses plagiarised documents in the corpora.

5. Discussion and Conclusions

We find that the Ferret can be effectively used to detect copied passages in Chinese text. The work described here is based on trigrams, but this was determined for English and the effect of using longer sequences should be investigated in future. Though the dictionary strategy will be slower and more complex, it will also be interesting to see how it performs.

Three strategies were investigated. The results from these experiments indicate that typeText performs better than typeChinese and typeMix. However, the test data had artificially produced plagiarism

which would be expected to do better on typeText than naturally occurring plagiarism. The reason for this is that pseudo-plagiarism is produced by copying entire passages, so there will be more matches of the long, multi-character tokens used in typeText. In the real world we usually find there are minor alterations and rewordings in an attempt to avoid detection. A single change in a string will mean there will be no match between two similar strings, even if parts are in fact the same. In this case the long tokens used for typeText would not be as useful as the other strategies.

In some real world situations typeText will be the most appropriate approach, for instance in comparing different versions of regularly revised reports, where there is no intention to deceive.

When typeText detects copying, we can be confident it exists: however, there may be copied text that it will miss which the finer-grained, single character strategy can find. In situations where there is a deliberate attempt to deceive, typeChinese and typeMix will be more robust than typeText, and are good enough to detect copied material up to the limits discussed above.

1. The single character strategy works well on Chinese documents for detecting real plagiarism. A typical optimum threshold of Ferret is round 0.04 to 0.05 for this data, when Chinese documents are treated as typeChinese or typeMix.

2. Where there is no attempt to deceive, or with pseudo-plagiarised documents, typeText is an effective strategy. A typical optimum threshold is round 0.01 to 0.02

3. The optimum threshold for any particular corpus can be found by analysing a small sample of document pairs.

4. A higher threshold can increase precision but lose some potential plagiarised documents. The level of recall depends on the amount of copied material, and small amounts may not be detected. The typical lower limit of Ferret's detection ability is about 0.05 copy ratio. If the copied content is above this, then Ferret has a high probability of finding it.

5. Ferret is fast. The corpus Xi04 with about 1.3 million Chinese characters was processed in a few minutes on a standard desk top PC with 1G memory, 2.09 GHz, for all of the three algorithms.

By taking Chinese characters as tokens we depart from any semantic representation. A character will often be a part of a word and a trigram of characters may be devoid of meaning. It is in this sense that we use a

sub-symbolic representation, and observe the contrast between machine based engineering approaches and human based cognitive processing.

Acknowledgements

Dr. JunPeng Bao is working at the University of Hertfordshire, UK, sponsored by the Royal Society as a Visiting International Fellow.

References

Bao, J. P., J. Y. Shen, X. D. Liu, and H. Y. Liu: 2006, 'A fast document copy detection model'. *Soft Computing* **10**, 41–46.

Bao, J. P., C. Lyon, P. C. R. Lane, W. Ji, J. A. Malcolm: 2006, 'Copy detection in Chinese documents using the Ferret: a report on experiments'. Technical report 456: School of Computer Science, University of Hertfordshire.

Bao, J. P., J. Y. Shen, X. D. Liu, H. Y. Liu, and X. D. Zhang: 2004a, 'Finding plagiarism based on common semantic sequence model'. In: *Proceedings of the 5th International Conference on Advances in Web-Age Information Management*, pp. 640–645.

Broder, A. Z.: 1998, 'On the resemblance and containment of documents'. In: *Proceedings of Compression and Complexity of Sequences*, pp. 21–29.

Gao, J., M. Li, A. Wu, and C. N. Hang: 2006, 'Chinese word segmentation and named entity recognition: A pragmatic approach'. *Computational Linguistics* **31**, 531–573.

Giles, J: 2006, 'Preprint analysis quantifies scientific plagiarism'. *Nature* **444**, 524–525.

Lane, P. C. R., C. Lyon, and J. A. Malcolm: 2006, 'Demonstration of the Ferret plagiarism dectector'. In: *Proceedings of the 2nd International Plagiarism Conference.*

Lyon, C., R. Barrett, and J. A. Malcolm: 2003, 'Experiments in plagiarism detection'. Technical report 388: School of Computer Science, University of Hertfordshire.

Lyon, C., J. A. Malcolm, and R. G. Dickerson: 2001, 'Detecting short passages of similar text in large document collections'. In: *Proceedings of Conference on Empirical Methods in Natural Language Processing.*

Lyon, C., R. Barrett, and J. A. Malcolm: 2006, 'Plagiarism is easy, but also easy to detect'. *Plagiary* **1**, 1–10

Malpohl, G.: 2006 *JPlag: Detecting Software Plagiarism* `http://wwwipd.ira.uka.de:2222/`

Manning, C. D. and H. Schütze: 2001, *Foundations of statistical natural language processing.* Cambridge, MA: The MIT Press.

Turnitin: 2006 *Plagiairism Prevention* `http://www.turnitin.com`