International Journal of u-and e-Service, Science and Technology
Vol.7, No.4 (2014), pp.385-396
http://dx.doi.org/10.14257/ijunnesst.2014.7.4.35

Detecting High Obfuscation Plagiarism: Exploring Multi-Features Fusion via Machine Learning

Leilei Kong[1], Zhimao Lu[2], Haoliang Qi[3] and Zhongyuan Han[4]

[1]Harbin Engineering University, Heilongjiang Institute of Technology
[2]Harbin Engineering University
[3,4]Heilongjiang Institute of Technology
[1]kongleilei1979@gmail.com, [2]zhimaolu@163.com, [3]haoliangqi163@163.com,
[4]hanzhongyuan@gmail.com

Abstract

Providing effective methods of identification of high-obfuscation plagiarism seeds presents a significant research problem in the field of plagiarism detection. The conventional methods of plagiarism detection are based on single type of features to capture plagiarism seeds. But for high-obfuscation plagiarism detection, these single type features are not sufficient for identifying the plagiarism seeds effectively because of the varied plagiarism methods used in high-obfuscation plagiarism. This paper presents a multi-features fusion method for the high-obfuscation plagiarism seeds identification. This method exploits Logical Regression model to integrate lexicon features, syntax features, semantics features and structure features which extracted from suspicious document and source document. A multi-feature fusion classifier based on Logical Regression model is proposed to decide whether a text fragment pair can be regarded as plagiarism seeds or not. Experimental results on the PAN@CLEF2013 summary-obfuscation corpus show that the fusion of different types of features produces more accurate results.

Keywords: *plagiarism detection; high-obfuscation plagiarism; plagiarism seeds; multi-feature fusion; logical regression model*

1. Introduction

Plagiarism and its automatic retrieval have attracted considerable attention from research to industry: various papers have been published on the topic, and many commercial software systems are being developed [1]. For escaping the checks of plagiarism detection (PD) software, plagiarists begin using high obfuscation methods to plagiarize, for example, applying summarizing and paraphrasing approaches to carry out plagiarism. This type of plagiarism is called high-obfuscation plagiarism in international plagiarism detection evaluation which is organized by PAN@CLEF in recent years [2].

Many plagiarism detection methods focus on capturing the no-obfuscation and low-obfuscation plagiarism. In fact, most of the existing systems fail to detect plagiarism by paraphrasing the text, by summarizing the text but retaining the same idea [3]. High-obfuscation plagiarism has become one of the difficult problems in PD and it is a plagiarism way which is most difficult to be detected. Investigating its reason, the high-obfuscation plagiarism are obfuscated by reduction, combination, paraphrasing, summarizing, restructuring, concept specification and concept generalization to modify the text and change most of its appearance. This makes plagiarism methods even more difficult to capture the

ISSN: 2005-4246 IJUNNEST

possible plagiarism fragments between the two documents which we call them plagiarism seeds. Capturing the high-obfuscation plagiarism seeds is one of the key tasks of plagiarism detection and its recall will have an immense effect on the performance of PD. By coming up with as many reasonable seeds as possible, the subsequent step of “growing” them into aligned passages of text becomes a lot easier [4].

Textual features are essential to capture different types of plagiarism. At present most methods do not distinguish the type of plagiarism. The methods based on textural lexical features are the most critical in the process of searching plagiarism seeds. Character-based n-gram, word-based n-gram and vector-based features are commonly used by many of the existing plagiarism detection approaches. These approaches use string match, fingerprint or words frequency statistics to identify the similarity between the two text fragments to obtain the plagiarism seeds. For example, in [5] an ENCOPLOT system is described which uses char 16-grams, [6] word 8-gram matching, and [7] word 5-gram matching, to look for plagiarized seeds. These approaches are fast but at the expense of losing semantic information. Losing large numbers of plagiarism seeds always results in a low performance for high-obfuscation plagiarism detection because we cannot identify a mass of high-obfuscation plagiarism segments when we only use the methods based on lexical features to detect plagiarism.

Recently, as plagiarism detection is investigated further, the researchers has realized that the plagiarism seeds would be lost if only make use of lexical features to detect plagiarism. The researchers attempt to use other textual features for plagiarism detection and more and more new methods are proposed constantly. [8, 9] focused on syntactic features, decomposing documents into statements and extracting part of speech (POS) features, to detect the plagiarism by POS of phrases and words in different statements. [10, 11] used semantic features considering quantify the use of word classes, synonyms, antonyms, hyponyms, and hyponyms, incorporating different semantic features, thesaurus dictionaries, and lexical databases, such as WordNet to identify plagiarism seeds. However the method based on semantic features have received less attention in plagiarism detection research due to the challenge of representing semantics and the time complexity of representative algorithms. The structural features of the text fragments were also considered in some researches. For example, [12] applied stopwords sequence and [13] employed a coarse-to-fine framework to detect plagiarism. By using structure, contextual information (i.e., topical blocks, sections, and paragraphs) was taken into account, which carries different importance of text, and characterizes different ideas distributed throughout the document.

Implementing rich feature structures should lead to the detection of more types of plagiarism [3]. However, no successful experiments have been performed in a public corpus to prove the performance of all kinds of these features and integrated them to obtain more high-quality plagiarism seeds. To this article, the forms of high-obfuscation plagiarism are numerous, intuitively, the various features should be combined to discriminate the different styles plagiarism. Combining various kinds of features to search plagiarism seeds is an important ways to improve the performance of high-obfuscation plagiarism detection.

Our methods use Logical Regression Model of discriminative learning method to integrate the lexical, syntactic, semantic and structure features. The model will learn a set of weights for each feature to get a classifier for identifying the high-obfuscation plagiarism seeds. We choose the method [14] as baseline which got the first place in the plagiarism detection evaluation organized by PAN@CLEF2012. This paper shows the effect when using the single textual feature firstly. Then it shows the results of multi-features fusion by using Logical Regression Model. The results verified that multi-features fusion method can significantly improve the performance of plagiarism detection.

2. Related Work

Plagiarism detection can be defined as follows [1]: Let s=(s_{plg}, d_{plg}, s_{src}, d_{src}) denote a plagiarism case where s_{plg} is a passage of document d_{plg} and a plagiarized version of some source passage s_{src} in d_{src}. Given d_{plg}, the task of a plagiarism detection is to detect s by reporting a corresponding plagiarism detection r=(r_{plg}, d_{plg}, r_{src}, d'_{src}). The process which r detects s iff $s_{plg} \cap r_{plg} \neq \Phi$, $s_{src} \cap r_{src} \neq \Phi$, and $d_{src}=d'_{src}$ is called text alignment.

Martin Potthast proposed the main framework of text alignment [4]: (1) seeding, (2) match merging, and (3) extraction filtering. Given a suspicious document d_{plg} and a source document d_{src}, identifying all the possible plagiarism fragments of document d_{plg} and d_{src} is core issue of text alignment of plagiarism detection. These fragments are called *plagiarism seeds*. The framework of text alignment shows in Figure 1 and the inside of dashed rectangle is our research work:

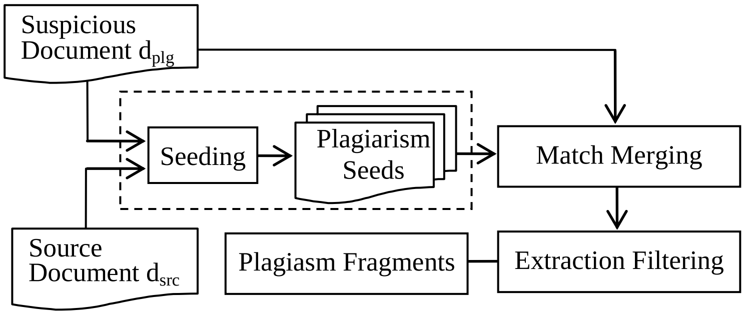

Figure 1. Framework of Text Alignment

For high ambiguity plagiarism detection, obtaining the high-quality seeds is the key step and it is the most difficult problem to solve. The statistical data based on PAN@CLEF2012 training corpus found that the averaged Jaccard coefficient of high-obfuscation plagiarism fragments is 0.2424, well below the 0.9968 of low-obfuscation plagiarism. Therefore, the detection methods which obtain a good performance in the no-obfuscation and low-obfuscation plagiarism detection do not achieve the satisfactory results in the detection of high-obfuscation plagiarism.

In the process of searching plagiarism seeds, lexical features, syntactic features, semantic features and structural features are commonly used by the researchers.

Earlier researches have demonstrated the effectiveness of lexical features in the text alignment task of plagiarism detection. Lexical features operate at the character or word level. Lexical features which are usually based on characters or words n-gram are the most common ways. Words n-gram may be provided by a simple 2-grams, 3-grams or n-grams. These words n-grams are called fingerprint or shinglings. The approaches of plagiarism detection obtained the plagiarism degree of text fragments pairs by using the common text similarity calculation method to get the fingerprints of suspicious document and source document. Consequently, the plagiarism seeds are found by this way. The researches based on lexical features include character-based or words-based string matching methods [8], fingerprints or words frequency statistics methods [5, 6, 7], vector-based methods [15, 16, 17, 18] and so on.

Syntactic features are also used in plagiarism detection. Existing methods are mainly using Shallow Parsing, which take the part of speech (POS) of phrases or sentences as the feature. Goman[19]proved that the combination of the syntactic features and lexical features can improve the results of rewrite rule frequencies. In recent studies, Elhadi and Al-Tobi [8, 9]used POS tags features followed by other string similarity metrics in the analysis and calculation of similarity between texts.

Synonyms, antonyms, replacement, concept generalization and specialization are commonly used by plagiarists in high-obfuscation plagiarism. If detection methods use lexical features to detect the plagiarism which is carried out by the way of synonyms replacement, they will misjudge the plagiarism degree of the two text fragments because the methods based on lexical features cannot dig out the semantic information. By this means, the plagiarism seeds are lost. Semantic features quantify the use of word classes, synonyms, antonyms, hyponyms, and hyponyms. The use of thesaurus dictionaries and lexical databases, WordNet, for instance, would significantly provide more insights into the semantic meaning of the text. Together with POS tagging, semantic dependencies can be featured, and that would be very helpful in plagiarism detection [3].

Stamatatos [12] proved that using the information of syntactic structure of sentences tends to be more reliable in plagiarism detection. Stamatatos pointed out in [12], stopwords occurrences are usually associated with syntactic patterns. Therefore, sequences of stopwords reveal hints of the syntactic structure of the document that is likely to remain stable during the procedure of plagiarizing a passage. The method of Stamatatos take two text fragments stopwords n-gram as structural features to detect plagiarism. In a recent study, [13] identify the plagiarism by using a document tree structure. Structure features methods have been used to detect copy-and-paste plagiarism.

Different feature may be utilized to detect different types of plagiarism. For high-obfuscation plagiarism detection, deeper understanding the effects of all types of features will contribute to better plagiarism detection performance. Further research should be carried out to integrate the advantages of relating lexical features with syntactic, semantic and structure features for plagiarism detection. Features can be combined by statistical learning theory. However, using multi - feature fusion to judge the plagiarism is still not the position of mainstream ideology and the related works are quite rare.

This paper uses the Logical Regression Model to combine the lexical, syntactic, semantic and structure features to detect high-obfuscation plagiarism and hope to capture more plagiarism seeds by using the characteristic of all types of features.

3. Multi-features fusion via Logical Regression model

3.1. Problem

As in this heading, they should be Times New Roman 11-point boldface, initially capitalized, flush left, with one blank line before, and one after. Let d_{plg} = (s_1, s_2, ..., s_n), dsrc = (r_1, r_2, ..., r_n), which s_i and r_j are the text fragments of d_{plg} and d_{src}. Calculate the various types of features between the two text fragments pair, then we can obtain a features set of (s_i, r_j) which is composed by their similarity degree scores. Set it as ($f_{i,j}^{1}$, $f_{i,j}^{2}$,......, $f_{i,j}^{n}$), which $f_{i,j}^{k}$ is the k-th feature value of text fragment pair (s_i, r_j). These features are probabilistically independent. For determining whether the text fragments pair is plagiarism or not, our goal is to get the plagiarism probability of (s_i, r_j) by considering the features set ($f_{i,j}^{1}$, $f_{i,j}^{2}$,......, $f_{i,j}^{n}$). These suspicious text fragments will be regarded as plagiarism seeds. Finally, r =(r_{plg}, d_{plg}, r_{src}, d'_{src}) will be gained by the process of match merging, and extraction filtering. In literature, there are many methods concerning the estimation of the probability for a given features set.

The discriminative learning method can directly learn the conditional probability P (Y|X) or the decision function f (X), so we choose discriminative learning method to estimate the weight of each feature based on training data. The model will be used to get the final plagiarism probability of (s_i, r_j). Regarding the successful application of the Logical Regression Model in the field of text classification and the sufficient training data, this paper selected the Logical Regression Model to integrate the multiple types features to detect the high-obfuscation plagiarism. The plagiarism probability of text fragment $s_i \in d_{plg}$ and $r_j \in d_{src}$ will be obtained by a classifier C which is trained based on binomial Logical Regression Model by using training dataset. If the probability of plagiarism is greater than the one of no plagiarism, the output of classifier is 1, otherwise, the output is 0. Furthermore, for given $d_{plg}=(s_1,s_2,\ldots\ldots,s_n)$ and $d_{src}=(r_1,r_2,\ldots\ldots,r_n)$, we can obtain a list of all plagiarism seeds by using the classifier C.

This process can be described as follows：

Training Phase:
Given: Training text fragment pairs and their label p=(0 or 1, s_i, r_j).
Output: weight($w^k_{i,j}$, $f^k_{i,j}$)and C.
Testing Phase:
Given: suspicious document d_{plg} and source document d_{src}.
Output: the list contains all plagiarism seeds.

3.2. Multi-Features Fusion

Logistic Regression model can effectively integrate all types of features. It will learn a set of weights for each feature of the text fragment pairs on training data. On test data, when a new text fragment pairs arrives, Logistic Regression model find this list of features and sum the weights associated with those feature. In mathematical features, we will write θ·x. In this notation, θ is a vector of weights, and x is a vector of features. The notation θ·x simply means to take the sum of weights associated with each of these features. We then convert this sum of weights to a probability, using the logistic functions (1) and (2) [20],

$$P(Y=1|x)=\frac{\exp(\theta\cdot \mathrm{x})}{1+\exp(\theta\cdot \mathrm{x})} \tag{1}$$

$$P(Y=0|x)=\frac{1}{1+\exp(\theta\cdot \mathrm{x})} \tag{2}$$

This simple equation converts a number between −∞ and +∞ to a probability between 0 and 1. If this probability P(Y=1|X) is greater than the probability P(Y=0|X), we predict that the text fragment pair is plagiarism. The gradient descent is a common method to solve θ.

3.3. Features set for high-obfuscation plagiarism detection

In this paper, features to measure the plagiarism degree of text fragment pairs include: *lexical features*, *syntactic features*, *semantic features* and *structure features*. We now describe these features for high-obfuscation plagiarism detection that we use.

3.3.1. Lexical features: The lexical feature which we used in this paper is word-based n-gram and character-based n-gram which show as follows.

Dice Coefficient: It counts how many common character s_i and r_j have. It is defined as:

$$DC(s_i,r_j)=\frac{2\cdot|s_i \cap r_j|}{|s_i|+|r_j|} \tag{3}$$

where, $|s_i \cap r_j|$ is the number of matching characters, $|s_i|+|r_j|$ is the total character number of s_i and r_j.

Jaro Distance: The Jaro distance [21] of s_i and r_j is defined as:

$$\mathrm{JD}\left(s_i, r_j\right)=\frac{m}{3\times l_1}+\frac{m}{3\times l_2}+\frac{m-t}{3\times m} \tag{4}$$

where, m is the number of characters of s_i that match characters of r_j, t is half the number of matching characters with different sequence, l_1 and l_2 are the lengths (in characters) of s_i and r_j, respectively.

Jaccard Coefficient: The raido of number of shared terms against total char number of terms. This is defined as:

$$\mathrm{JC}\left(s_i, r_j\right)=\frac{|s_i \cap r_j|}{|s_i \cup r_j|} \tag{5}$$

Levenshtein Distance: This is the minimum number of operations (edit distance) needed to transform one string (in our case, s_i) into the other one (r_j), where an operation is an insertion, deletion, or substitution of a single character.

Manhattan Distance: this is defined for any two vectors x and y in an n dimensional vector space as:

$$MD\left(\vec{x},\vec{y}\right)=\sum_{i=1}^{n}\frac{|x_i-y_i|}{|x_i|-|y_i|} \tag{6}$$

in our case, n is the number of distinct words that occur in s_i and r_j (in any of the two); and x_i, y_i show how many times each one of these distinct words occurs in s_i and r_j, respectively.

Ngram Distance: NGD [22] is the same as MD, but the terms is instead by distinct character ngrams in s_i and r_j; we used n = 1, 2 and 3.

Soundex Distance: Soundex is defined in [23] and we use the methods in [22] to get the Soundex.

3.3.2. Syntactic Features

POS n-gram Distance: This is defined as follows:

$$\mathrm{POSD}\left(s_i, r_j\right)=\frac{|\mathrm{POS}(s_i)\cap \mathrm{POS}(r_j)|}{|\mathrm{POS}(s_i)\cup \mathrm{POS}(r_j)|} \tag{7}$$

where the numerator is used to denote the number of matching POS n-gram, and the denominator is the sum of all POS n-gram in sentence pair (s_i,r_j). POS of a sentence is extracted by Stanford’s POS taggar [24].We used n = 3.

Noun Ratio: We found that proper noun modification in plagiarism is very rare. Noun Ratio is used to describe the similarity of proper noun. Proper noun could be a person, place, thing, etc. The definition of NRO follows:

$$\mathrm{NRO}\left(s_i, r_j\right)=\frac{\mathrm{NNP}\left(s_i, r_j\right)}{|s_i|+|r_j|} \tag{8}$$

where NNP (si, rj) represents the number of common proper noun in s_i and r_j. | s_i | and | r_j | are the terms number of s_i and r_j.

3.3.3. Semantic Features

Semantic Similarity Distance: The semantic similarity feature between s_i and r_j is designed as follows:

$$\mathrm{SSD}\left(s_i, r_j\right)=\frac{\left(\frac{\sum_{i=1}^{m} a_i}{\left|s_i\right|}+\frac{\sum_{i=1}^{n} b_j}{\left|r_j\right|}\right)}{2} \tag{9}$$

where $a_i=\max(D_s(a_i,b_1),\ D_s(a_i,b_2),\ \ldots\ldots,\ D_s(a_i,b_n))$, $b_j=\max(D_s(a_1,b_j),\ D_s(a_2,b_j),\ldots\ldots,D_s(a_n,b_j))$, Ds is the term a_i and b_j semantic similarity which is calculated using the method proposed by Leacock and Chodorow [^{2}5]. WordNet 3.0 is used as semantic database. $|s_i|$ and $|r_j|$ are as in the previous measure.

3.3.4. Structure Features

Stopword N-gram Distance: The methods proposed by Stamatatos [12] is used to get the structure similarity of the two sentences. Stopword n-gram distance is measured as follows:

$$\mathrm{g} \in P(n_1, d\mathrm{x}) \cap P(n1, d\mathrm{s}):\ member(g,C)<n1\text{-}1 \wedge\ maxseq(g,C)<n1\text{-}2 \tag{10}$$

where the functions member(g,C) and maxseq(g,C) return the number of stopwords of the n-gram g that belong to C and the maximal sequence of words of g that belong to C, respectively. And C={the, of, and, a, in, to, 's}. $P(n_1,d_x)$ and $P(n_1,d_s)$ are the corresponding profiles.

Word Pair Order: We also applied Word Pair Order proposed by Hatzivassiloglou[26]. This feature is used to calculate two words occur in the same order in both text fragments.

String Length Ratio: The ratio of minimum string's length to maximum string's length between s_i and r_j as:

$$\mathrm{SLR}\left(s_i, r_j\right)=\frac{\min\left(L_T, L_H\right)}{\max\left(L_T, L_H\right)} \tag{11}$$

4. Result

4.1. Corpus

Experimental data is 04_artificial_high, which is a sub corpus of PAN @ CLEF2012 training data for the task of detailed comparison (which is called Text Alignment in PAN@CLEF2013). Evaluating plagiarism detection algorithm is one of the main contents of PAN. [4] elaborated on the structure of the data set. A more detailed description of the obfuscation strategies can be found in [27]. Table 1 statistics the related information of 04_artificial_high sub-corpus.

Table 1. Corpus Statistics for Training Dataset

Suspicious Document Length	short (<10000words)	50.96%
	medium (10000-50000words)	28.95%
	long (>50000words)	20.09%
Source Document Length	short (<10000words)	45.00%
	medium (10000-50000words)	21.07%
	long (>50000words)	33.93%
Plagiarism Case Length	short (<150 words)	13.0%
	medium (150-1150 words)	39.1%
	long (>1150 words)	47.9%
Plagiarism per Document	hardly (3%-20%)	77.4%
	medium (20%-50%)	12.2%
	much (50%-80%)	8.6%
	entirely (>80%)	1.8%
Orerlapping Rate	short (<0.2)	42.93%
	medium (0.2-0.5)	46.66%
	Long(>0.5)	10.41%

In this paper, we take the two Summary-Obfuscation sub-corpus as the test data (we call them SO1 and SO2), which are the test subsets of PAN@CLEF2013 offered by PAN in the task of Text Alignment. Table 2 shows the statistics of the two test subsets.

Table 2. Corpus Statistics for Test Dataset

		SO1(%)	SO2(%)
Suspicious Document Length	short(<700words)	6.61%	6.08%
	medium (700-2000words)	89.25%	88.19%
	long (>2000words)	4.14%	5.73%
Source Document Length	short (<700words)	50.00%	47.68%
	medium (700-2000words)	46.08%	49.79%
	long (>2000words)	3.92%	2.53%
Plagiarism Case Length	short (<50 words)	0.00%	0.04%
	medium (50-100 words)	41.67%	28.39%
	long (100-150words)	58.33%	71.61%
Plagiarism per Document	hardly (3%-5%)	4.160%	5.08%
	medium (5%-10%)	41.67%	28.39%
	much (10%-15%)	54.17%	63.56%
	entirely (>15%)	0.00%	2.97%
Orerlapping Rate	short (<0.2%)	75.00%	85.17%
	medium (0.2-0.5%)	25.00%	14.83%
	Long(>0.5%)	0.00%	0.00%

4.2. Measures

For evaluating the produced detections, we use the recently proposed measures of macro-average *precision*, *recall* and *granularity* [1]. Likewise, a plagiarism detection r∈R is represented as r. Based on this notation, precision and recall of R under S can be measured as follows:

$$prec(S,R)=\frac{1}{|R|}\sum_{r\in R}\frac{|U_{s\in S}(s\cap r)|}{|r|} \qquad (12)$$

$$rec(S,R)=\frac{1}{|S|}\sum_{s\in S}\frac{|U_{r\in R}(s\cap r)|}{|s|} \quad (13)$$

where if r detects then s∩r is s∩r, otherwise, s∩r is Φ.

A detector’s granularity is quantified as follows:

$$gran(S,R)=\frac{1}{|S_R|}\sum_{s\in S_R}|R_s| \quad (14)$$

where S_R⊆S are cases detected by detections in R, and Rs ⊆R are detections of s; i.e., SR = {s|s∈ S∧ ∃r∈ R : r detects s} and Rs = {r∈ R∧ r detects s}. The measures are combined into a single overall score as follows:

$$pldget(S,R)=\frac{F_1}{\log_2(1+gran(S,R))} \quad (15)$$

where F_1 is the equally weighted harmonic mean of precision and recall.

4.3. Baseline

The methods proposed in [14] gain the first place on the measures of Overall Score and Artificial High sub-corpus in the evaluation of the Detailed Comparison task in PAN@CLEF2012. We choose it as our baseline. This method split dplg and dsrc into sentences and compute the Cosine similarity to get the first step plagiarism seeds. Cosine similarity is defined as follows:

$$sim(S,R)=\cos\theta=\frac{\sum_{k=1}^{n}W_{S_k}*W_{R_k}}{\sqrt{(\sum_{k=1}^{n}{W_{S_k}}^2)(\sum_{k=1}^{n}{W_{R_k}}^2)}}>t_1 \quad (16)$$

where sim(S,R) is the similarity degree of sentence S and R, θ is a document vector angel, WSk and WRk are the weight of S and R respectively, t1 is a threshold and equals to 0.415.

Then, the baseline filter the plagiarism seeds by using the Jaccard Coefficient which defined as follows:

$$T=\frac{2*\sum_{t\in I_s\cap I_R}Min(N_{I_S}(t),N_{I_R}(t))}{|I_s|+|I_R|}>t_2 \quad (17)$$

where NIs(t) and NIR(t) are the number of the terms which are overlapping in the suspicious sentence and reference sentence, Min(NIs(t),NIR(t)) is the smallest one of NIs(t) and NIR(t) , t2 is the threshold and it is 0.32.

4.4. Experimental results

Experiment is done based on the Baseline (Kong2013). Formula (16) is used to get the first-step plagiarism seeds and all features as described above are used to replace (17) to verify the effect of a single feature. t_2 is set according to the best performance in PAN@CLEF2013 training corpus. The test corpuses are the two *05-summary-obfuscation* plagiarism sub-corpus of PAN@CLEF2013. The results are shown in Table 3, Table 4. Our method is marked by *M-feature*.

Table 3. Comparison Results of Single-Feature via Multi-Feature Fusion on SO1

	Features	Plagdet	Precise	Recall	Granularity
1	Kong2013	0.58120	0.95703	0.41731	1.00000
2	Dice Coefficient	0.56136	0.95421	0.39765	1.00000
3	Levenshtein Distance	0.56421	0.95375	0.40060	1.00000
4	Manhattan Distance	0.56868	0.97139	0.40202	1.00000
5	Ngram Distance	0.43290	0.97091	0.27855	1.00000
6	Soundex Distance	0.47708	0.97944	0.31534	1.00000
7	POS n-gram Distance	0.57583	0.95392	0.41238	1.00000
8	Noun Ratio	0.52144	0.98185	0.35498	1.00000
9	Semantic Similarity Distance	0.47355	0.94780	0.33895	1.07692
10	Stopword N-gram Distance	0.33560	**0.99939**	0.20165	1.00000
11	Word Pair Order	0.48918	0.94417	0.33011	1.00000
12	String Length Ratio	0.55961	0.95420	0.39589	1.00000
13	M-feature	**0.72563**	0.94167	**0.59022**	1.00000

Table 4. Comparison Results of Single-Feature via Multi-Feature Fusion on SO2

	Features	Plagdet	Precise	Recall	Granularity
1	Kong2013	0.43399	0.96381	0.30016	1.07742
2	Dice Coefficient	0.43940	0.96391	0.30522	1.07792
3	Levenshtein Distance	0.44724	0.96079	**0.30933**	1.06536
4	Manhattan Distance	0.42389	0.96467	0.29099	1.07741
5	Ngram Distance	0.42048	0.96216	0.28408	1.06081
6	Soundex Distance	0.33560	0.95854	0.21055	1.04032
7	POS n-gram Distance	0.43984	0.96141	0.30440	1.07237
8	Noun Ratio	0.44452	0.96305	0.30496	1.05921
9	Semantic Similarity Distance	0.42508	0.96676	0.29014	1.07051
10	Stopword N-gram Distance	0.36922	0.97830	0.24107	1.06716
11	Word Pair Order	0.12423	**0.99955**	0.06623	1.00000
12	String Length Ratio	0.41696	0.96978	0.28168	1.06623
13	M-feature	**0.43681**	0.96154	0.30336	1.07895

All types of plagiarism features are also used to integrated to compare with the M-feature for demonstrating the validity. Table 5 and Table 6 show the results of lexical, syntactic, semantic and structural features via M-feature on SO1 and SO2.

Table 5. Comparison results of single-type-feature via multi-feature fusion on SO1

	Features	Plagdet	Precise	Recall	Granularity
1	M-feature	**0.72563**	0.94167	**0.59022**	**1.00000**
2	Lexical	0.69249	0.95076	0.57421	1.04761
3	Semantic	0.59751	0.93349	0.46539	1.05555
4	Syntactic	0.60081	0.84234	0.51751	1.09523
5	Structure	0.60337	**0.96798**	0.43829	**1.00000**

Table 6. Comparison results of single-type-feature via multi-feature fusion on SO2

	Features	Plagdet	Precise	Recall	Granularity
1	M-feature	**0.58614**	0.89394	**0.46312**	1.05759
2	Lexical	0.58175	**0.89818**	0.44223	1.02617
3	Semantic	0.42169	0.89733	0.29917	1.09090
4	Syntactic	0.54317	0.87405	0.43511	1.09890
5	Structure	0.54470	0.86980	0.43689	1.09625

Experiments indicate that the recall increase 41.43% and 54.29%, and the total score plagdet increase 24.85% and 35.05% than the baseline which gained the first place in PAN@CLEF2013. Multi-feature fusion on single textual feature type can also improve the measure of *plagdet*. There is obvious superiority sub-type of features fusion than the single feature. The results show that the lexical features fusion is better than the other type of features fusion, followed by structure features fusion and syntactic features fusion. The detection performance of semantic features fusion is better when compared to single semantic feature but lower than other types of features fusion. In short, multi-feature fusion can offers higher *recall* to improve the total score *plagdet*.

5. Conclusion

The identification of high-obfuscation plagiarism seeds is one of the most difficult problems to be solved in plagiarism detection. Single type features cannot identify the plagiarism seeds effectively because of the varied plagiarism methods which used in high-obfuscation plagiarism. The characteristic of artificial-high plagiarism make it necessary for using more types of features to detect the plagiarism. Using different features to obtain different kinds of plagiarism can capture more plagiarism seeds. This paper integrates the lexical features, syntactic features, semantic features and structural features extracted from suspicious text fragments pairs. Our method integrates all types of features by using Logical Regression model with simplicity and effect. With the best fusion methods, the multiple feature fusion gave substantively improved plagiarism seeds recognition recall compared with traditional single-feature recognition.

Acknowledgements

This paper is supported by the Nation Natural Science Foundation of China(No.61370170), the Natural Science Foundation of Heilongjiang Province(No. F201224), the Heilongjiang Institute of Technology Youth Fund(No. 2012QJ11). Communication author: Haoliang Qi.

References

[1] Potthast M, Eiselt A, Barrón-Cedeno A, et al. Overview of the 3rd International Competition on Plagiarism Detection[C]//CLEF (Notebook Papers/Labs/Workshop). 2011.

[2] www.pan.webis.de

[3] Alzahrani S M, Salim N, Abraham A. Understanding plagiarism linguistic patterns, textual features, and detection methods[J]. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 2012, 42(2): 133-149.

[4] Potthast M, Gollub T, Hagen M, et al. Overview of the 4th International Competition on Plagiarism Detection[C]//CLEF (Online Working Notes/Labs/Workshop). 2012.

[5] Grozea C, Gehl C, Popescu M. ENCOPLOT: Pairwise sequence matching in linear time applied to plagiarism detection[C]//3rd PAN Workshop. Uncovering Plagiarism, Authorship and Social Software Misuse. 2009: 10.

[6] C. Basile, D. Benedetto, E. Caglioti, G. Cristadoro, and M. D. Esposti. A plagiarism detection procedure in three steps: Selection, matches and squares. *Proc. SEPLN*, Donostia, Spain, pp. 19–23.

[7] Kasprzak J, Brandejs M, Kripac M. Finding plagiarism by evaluating document similarities[C]//Proc. SEPLN. 2009, 9: 24-28.

[8] Elhadi M, Al-Tobi A. Duplicate detection in documents and webpages using improved longest common subsequence and documents syntactical structures[C]//Computer Sciences and Convergence Information Technology, 2009. ICCIT'09. Fourth International Conference on. IEEE, 2009: 679-684.

[9] Elhadi M, Al-Tobi A. Use of text syntactical structures in detection of document duplicates[C]//Digital Information Management, 2008. ICDIM 2008. Third International Conference on. IEEE, 2008: 520-525.

[10] S. Alzahrani and N. Salim. Fuzzy semantic-based string similarity for extrinsic plagiarism detection: Lab report for PAN@CLEF10, presented at the 4th Int. Workshop PAN-10, Padua, Italy, 2010.

[11] A. J. A. Muftah. Document plagiarism detection algorithm using semantic networks. M.Sc. thesis, Faculty Comput. Sci. Inf. Syst., Univ.Teechnol. Malaysia, Johor Bahru, 2009.

[12] Stamatatos E. Plagiarism detection using stopword ngrams[J]. Journal of the American Society for Information Science and Technology, 2011, 62(12): 2512-2527.

[13] Zhang H, Chow T W S. A coarse-to-fine framework to efficiently thwart plagiarism[J]. Pattern Recognition, 2011, 44(2): 471-487.

[14] Leilei Kong, Haoliang Qi, Shuai Wang, Cuixia Du, Suhong Wang, and Yong Han. Approaches for Candidate Document Retrieval and Detailed Comparison of Plagiarism Detection—Notebook for PAN at CLEF 2012. In Forner et al. [6].ISBN 978-88-904810-3-1.

[15] M. Murugesan, W. Jiang, C. Clifton, L. Si, and J. Vaidya. Efficient privacy-preserving similar document detection. *VLDB J.*, vol. 19, no. 4,pp. 457–475, 2010.

[16] Lyon C, Malcolm J, Dickerson B. Detecting short passages of similar text in large document collections[C]//Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing. 2001: 118-125.

[17] White D R, Joy M S. Sentence-based natural language plagiarism detection[J]. Journal on Educational Resources in Computing (JERIC), 2004, 4(4): 2.

[18] Barrón-Cedeño A, Basile C, Degli Esposti M, et al. Word length n-Grams for text re-use detection[M]//Computational Linguistics and Intelligent Text Processing. Springer Berlin Heidelberg, 2010: 687-699.

[19] Gamon M. Linguistic correlates of style: authorship classification with deep linguistic analysis features[C]//Proceedings of the 20th international conference on Computational Linguistics. Association for Computational Linguistics, 2004: 611.

[20] Li Hang. Statistical Learning Theory. Tsinghua University press,2012.

[21] Jaro M A. Probabilistic linkage of large public health data files[J]. Statistics in medicine, 1995, 14(5- 7): 491-498.

[22] Malakasiotis P, Androutsopoulos I. Learning textual entailment using SVMs and string similarity measures[C]//Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. Association for Computational Linguistics, 2007: 42-47.

[23] http://en.wikipedia.org/wiki/Soundex.

[24] http://nlp.stanford.edu/software/tagger.shtml

[25] Leacock C, Chodorow M. Combining local context and WordNet similarity for word sense identification[J]. WordNet: An electronic lexical database, 1998, 49(2): 265-283.

[26] Hatzivassiloglou V, Klavans J L, Eskin E. Detecting text similarity over short passages: Exploring linguistic feature combinations via machine learning[C]//Proceedings of the 1999 joint sigdat conference on empirical methods in natural language processing and very large corpora. 1999: 203-212.

[27] Martin Potthast. Technologies for Reusing Text from the Web. Dissertation, Bauhaus-Universität Weimar, December 2011.

Author

Leilei Kong, born in 1979, Ph. D. candidate, lecturer. Her research interests include plagiarism detection, information retrieval, and natural language processing.