Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 111 (2017) 382–387

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Conference on Advances in Information Technology, IAIT2016, 19-22 December 2016, Macau, China

Plagiarism detection using document similarity based on distributed representation

Kensuke Baba[a,*], Tetsuya Nakatoh[b], Toshiro Minami[c]

[a]*Fujitsu Laboratories, Kawasaki, Japan*
[b]*Kyushu University, Fukuoka, Japan*
[c]*Kyushu Institute of Information Sciences, Dazaifu, Fukuoka, Japan*

Abstract

Accurate methods are required for plagiarism detection from documents. Generally, plagiarism detection is implemented on the basis of similarity between documents. This paper evaluates the validity of using distributed representation of words for defining a document similarity. This paper proposes a plagiarism detection method based on the local maximal value of the length of the longest common subsequence (LCS) with the weight defined by a distributed representation. The proposed method and other two straightforward methods, which are based on the simple length of LCS and the local maximal value of LCS with no weight, are applied to the dataset of a plagiarism detection competition. The experimental results show that the proposed method is useful in the applications that need a strict detection of complex plagiarisms.

Peer-review under responsibility of the organizing committee of the 8th International Conference on Advances in Information Technology

Keywords: Plagiarism detection, document similarity, longest common subsequence, distributed representation

1. Introduction

Accurate methods are required for plagiarism detection with a huge amount of document data. With the spread of computers and the Internet, a large number of documents became available as electronic data. Digital documents are easy to copy and reuse, which encourages plagiarisms from copyrighted contents and academic documents such as

* Corresponding author. Tel.: +81-44-754-2328.
E-mail address: baba.kensuke@jp.fujitsu.com

1877-0509
Peer-review under responsibility of the organizing committee of the 8th International Conference on Advances in Information Technology
10.1016/j.procs.2017.06.038

research papers. This situation impedes the sound development of the creative activities of humans. A simple solution for the problem is to develop a method that detects plagiarisms from a large number of documents as accurately as possible.

Plagiarism detection from documents can be formalized as a problem to compute a similarity of documents. Lukashenko et al.[4] summarized related studies to plagiarism detection and indicated that a viewpoint for classifying plagiarism detection methods is the measure of similarity between documents. An approach is using statistics of word occurrences such as the bag-of-words model[5]. Another approach is using patterns of word occurrences, such as the edit distance[10] and its weighted and local version[8] which are bases of sequence alignment in bioinformatics. A difficulty in applying the pattern matching-based approach to plagiarism detection in general documents exists on setting the similarity between words.

We propose a plagiarism detection method that uses a distributed representation of words[2] for setting a similarity between words. A distributed representation is regarded as a function that maps a word to a vector with a small dimension, and the distance between vectors represents a similarity between the words that correspond to the vectors. A simple distributed representation is available by reducing the dimension of a straightforward vector representation based on word frequency[5]. The recent work[6] in neural networks made easy to achieve a distributed representation that represents word similarity well from actual document data.

The aim of our study is evaluating the validity of using the distributed representation to define the word similarity for plagiarism detection. We introduce three methods based on the following three document similarities: for two documents,

- The length of the longest common subsequence (LCS)[1] divided by the length of the shorter document,
- The local maximal value of the length of LCS, and
- The local maximal value of the weighted length of LCS.

We propose a plagiarism detection method based on the last similarity. The proposed method corresponds to the sequence similarity computed by the Smith-Waterman algorithm[8]. Although there already exist plagiarism detection methods based on the algorithm[9,3], the novelty of the proposed method is using a distributed representation for the word similarity. Practically, the distributed representation was obtained from no particular data by word2vec[6]. We applied these three methods to the dataset for a competition of plagiarism detection[7] and investigated the accuracy of the plagiarism detection.

2. Methods

This section defines three methods for plagiarism detection. These methods use three kinds of document similarity, respectively. This section also defines the accuracy of plagiarism detection and describes the dataset for evaluating the methods.

2.1. Document Similarity

A document is defined to be a list of words. For a document d, $|d|$ is the length of d, that is, the size of the list. For $1 \leq i \leq |d|$, d_i is the ith word of d. For $1 \leq i < j \leq |d|$, $d_{i,j}$ is the document $(d_i, d_{i+1}, \dots, d_j)$. For $i < j$, $d_{i,j}$ is the empty document ε of length 0. For documents p and q, pq is the document obtained by combining the lists of p and q.

Let Dw for a document set D and a word w be the set of the documents dw for all d in D if D is not empty, and $\{w\}$ otherwise. Let $\mathrm{longest}\{D, E\}$ for document sets D and E be the set of the documents that are the longest in the union of D and E. Then, the set $L(p, q)$ of the *longest common subsequences* (*LCS*) between documents p and q is defined recursively as

$$L(p_{1:i}, q_{1:j}) = \begin{cases} \{\varepsilon\} & i = j = 0 \\ L(p_{1:i-1}, q_{1:j-1})p_i & p_i = q_j \\ \text{longest}\{L(p_{1:i-1}, q_{1:j}), L(p_{1:i}, q_{1:j-1})\} & p_i \neq q_j \end{cases} \quad (1)$$

We define three similarities between two documents for plagiarism detection methods.

The first one is the length of LCS divided by the length of the shorter document. This similarity $lcs(p, q)$ between documents p and q is defined to be $lcs'(p, q)/\min\{|p|, |q|\}$, where $lcs'(p, q)$ is, by the definition in Equation 1, $lcs'(p_{1:0}, q_{1:0}) = 0$ and

$$lcs'(p_{1:i}, q_{1:j}) = \max\{lcs'(p_{1:i-1}, q_{1:j-1}) + \delta(p_i, q_j), lcs'(p_{1:i-1}, q_{1:j}), lcs'(p_{1:i}, q_{1:j-1})\} \quad (2)$$

where $\delta(v, w)$ for words v, w is 1 if $v = w$, and 0 otherwise.

The second one is the length of a "local" LCS. The similarity $llcs(p, q)$ is defined as $llcs(p_{1:0}, q_{1:0}) = 0$ and

$$llcs(p_{1:i}, q_{1:j}) = \max\{llcs(p_{1:i-1}, q_{1:j-1}) + f(p_i, q_j), llcs(p_{1:i-1}, q_{1:j}) - 1, llcs(p_{1:i}, q_{1:j-1}) - 1, 0\} \quad (3)$$

where $f(v, w)$ for words v, w is 1 if $v = w$, and -1 otherwise.

The last one is the length of a "weighted" local LCS. The similarity $wllcs(p, q)$ is obtained from Equation 3 by replacing the f with a g that maps a pair of words to a real number in a range.

Table 1. The detailed numbers of the training and test data of plagiarism detection in PAN 2013

Label for plagiarism	Type of obfuscation	Number of pairs
Negative		1000 + 1000
Positive	No obfuscation	1000 + 1000
	Random obfuscation	1000 + 1000
	Translation	1000 + 1000
	Summarization	1185 + 1185

2.2. Plagiarism Detection and Accuracy

We define three plagiarism detection methods *lcs*, *llcs*, and *wllcs* on the basis of the three document similarities in Section 2.1, respectively. The input is a pair of documents and the output is "positive" (that is, there exists a plagiarism in a document from the other document) or "negative". Each method calculates one of the similarities between inputted documents, and then predict positive or negative by comparing the obtained similarity with a threshold.

Prior to the way to set the threshold, we define four measures of accuracy of plagiarism detection. Let tp, fp, tn, and fn be the numbers of the pairs predicated by a detection

- To be positive in the positive pairs (true positive),
- To be positive in the negative pairs (false positive),
- To be negative in the negative pairs (true negative), and
- To be negative in the positive pairs (false negative),

respectively. Then, the *accuracy*, the *precision*, and the *recall* of the detection are defined to be $(tp + tn)/(tp + fp + tn + fn)$, $tp/(tp + fp)$, and $tp/(tp + fn)$, respectively. The *F-measure* is defined to be the weighted harmonic mean of the precision and the recall.

We assume that some training data, that is, pairs of a pair of documents and a label of positive or negative, are given before conducting the plagiarism detection. The threshold is set with the training data so that the detection method satisfies one of the conditions

- ($p = r$): the precision is equal to the recall or
- ($r \geq 99.9$): the recall is equal to or more than 99.9%.

2.3. Dataset

To evaluate the plagiarism detection methods in Section 2.2, we applied the methods to a dataset for a competition, the text alignment task of plagiarism detection in PAN 2013[7], and investigated the four kinds of accuracy in Section 2.2. The dataset consists of training data and test data, and each dataset contains 4185 pairs of documents with a plagiarism (*positive pairs*) and 1000 pairs of documents with no plagiarism (*negative pairs*). Each set of positive pairs contains pairs of a plagiarism with no obfuscation, and with some types of obfuscation. The detail of the dataset is described in Table 1. The average length of the documents was about 1500.

Table 2. The accuracy (%) of the three plagiarism detection methods with two conditions for threshold

Condition	Method	Accuracy	Precision	Recall	F-measure
$p = r$	lcs	78.78	86.25	87.69	86.96
	llcs	**89.62**	90.28	**97.65**	**93.82**
	wllcs	87.38	**92.08**	92.30	92.19
$r \geq 99.9$	lcs	81.09	81.06	99.92	89.51
	llcs	81.83	81.63	**99.97**	89.88
	wllcs	**83.70**	**83.24**	99.92	**90.82**

We used word2vec for constructing distributed representation to implement the g for wllcs. For the learning of word2vec, we used the documents of the mentioned training data. Let $\beta(w)$ be the normalized vector applied to a word w by word2vec. Then, for words v and w, we set $g(v, w) = 2\langle\beta(v), \beta(w)\rangle - 1$. The similarity $g(v, w)$ is 1 if $v = w$. On the assumption that the expectation of the inner product is 0, the expectation of $g(v, w)$ for $v \neq w$ is nearly equal to −1 which is the value in llcs. The dimension of the vector $\beta(w)$ for any word w was set to 200.

3. Results

Table 2 shows the accuracy of the three plagiarism detection methods lcs, llcs, and wllcs with two conditions $p = r$ and $r \geq 99.9$ for setting the threshold. In the condition $p = r$, the accuracy of llcs was the best in the three methods. In the condition $r = 99.9$, wllcs gained the best accuracy, especially, the precision was better than llcs.

Table 3 shows the accuracy of the same methods for the dataset with the 1000 negative pairs and the 1000 positive pairs of plagiarisms with no obfuscation. In this case, the accuracies of llcs and wllcs were nearly 100%.

4. Discussion

We proposed a plagiarism detection method wllcs that uses distributed representation for a weight of word similarity. We discuss the validity of wllcs in comparison with llcs on the basis of the experimental results in Section 3.

4.1. Major Conclusion

The experimental results in Section 3 imply that the proposed method wllcs is a suitable for strict detection of complex plagiarisms. By the results in Table 2, wllcs achieved better accuracy than llcs in the case $r \geq 99.9$,

although wllcs could not gain better accuracy than llcs in the case $p = r$. The condition $r \geq 99.9$ aims that more than 99.9% of the plagiarisms can be detected. This situation is highly probable in actual applications of plagiarism detection. The strong point of wllcs is to be able to reduce the errors that a document with no plagiarism is predicted as a document with a plagiarism in this situation. Table 3 shows that llcs and wllcs can detect a simple plagiarism with extremely high accuracy, which means most errors are caused by plagiarisms with obfuscations.

Table 3. The accuracy (%) of the three plagiarism detection methods for a dataset with simple plagiarisms

Method	Accuracy	Precision	Recall	F-measure
lcs	82.95	83.31	82.40	82.85
llcs	99.90	**100**	99.80	99.89
wllcs	**99.95**	**100**	**99.90**	**99.94**

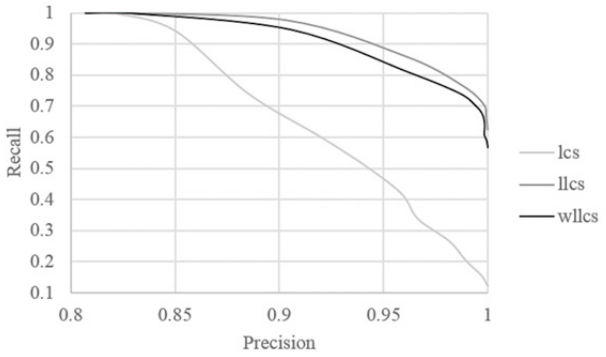

Fig. 1. Precision-recall graphs of the three plagiarism detection methods.

4.2. Key Findings

The superiority of wllcs to llcs is caused by the fact that wllcs treats a detailed similarity compared with llcs. Fig. 1 shows the precision-recall graphs for the three plagiarism detection methods. The trade-off of the two measure in llcs was better than that in wllcs. In spite of this result, some values in the results of wllcs were better than llcs in the case $r \geq 99.9$. The reason is that we can set a detailed threshold for the prediction because the document similarity wllcs is a real number while llcs can be only an integer. Actually, many pairs with no plagiarism and pairs with a plagiarism had the same similarity 2 in llcs.

4.3. Future Directions

Learning some parameters in the proposed method from training data is one of our future work. The computations of llcs and wllcs are same as that of the local alignment[8] in bioinformatics. We can regulate the penalties for a mismatch and an insertion/deletion which we set to -1 in this paper, and optimized parameters are expected to yield a better accuracy. The optimal penalties for wllcs depend on the distribution of the vector space $\beta(w)$ for words w. In addition to the penalties of a mismatch and an insertion/deletion, tuning the range and the distribution of $g(v, w)$ for words v and w will improve the accuracy.

Finding suitable corpus for training distributed representation is another one of our future work. In the computation of the distributed representation in wllcs, we used the given training data for learning by word2vec. Generally, it is expected that a larger dataset achieves a better accuracy. Investigating the effects of the corpus on plagiarism detection is meaningful in the case where extra knowledge is allowed to be used.

In this paper, we assumed plagiarism detection as an application of the proposed document similarity. We are going to apply the similarity to other applications such as analyses of rhetoric and finding adaptations of old poems.

5. Conclusion

We evaluated the validity of using distributed representation of words for defining a document similarity. We proposed a plagiarism detection method based on a document similarity with the weight defined by a distributed representation. We investigated the plagiarism detection accuracy of the proposed method and two straightforward methods with a dataset that includes several types of plagiarisms. The experimental results showed that the proposed method was superior to the other two methods in the case where an extremely high recall is required. Thus, we can conclude that distributed representation is applicable to a document similarity for plagiarism detection in the case where we need a strict detection of complex plagiarisms.

Acknowledgment

This work was supported by JSPS KAKENHI Grant Number 15K00310.

References

1. D. Gusfield. *Algorithms on Strings, Trees, and Sequences*. Cambridge University Press, 1997.
2. G. E. Hinton, J. L. McClelland, and D. E. Rumelhart. Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1. chapter Distributed Representations, pages 77-109. MIT Press, Cambridge, MA, USA, 1986.
3. R. W. Irving. Plagiarism and collusion detection using the smith-waterman algorithm. Technical report, 2004.
4. R. Lukashenko, V. Graudina, and J. Grundspenkis. Computer-based plagiarism detection methods and tools: An overview. In *Proceedings of the 2007 International Conference on Computer Systems and Technologies*, pages 1-6. ACM, 2007.
5. C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*. Cambridge University Press, 2008.
6. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111-3119, 2013.
7. M. Potthast, T. Gollub, M. Hagen, M. Tippmann, J. Kiesel, P. Rosso, E. Stamatatos, and B. Stein. Overview of the 5th International Competition on Plagiarism Detection. In *Working Notes Papers of the CLEF 2013 Evaluation Labs*, 2013.
8. T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *J. Mol. Biol.*, 147:195-197, 1981.
9. Z. Su, B.-R. Ahn, K.-Y. Eom, M.-K. Kang, J.-P. Kim, and M.-K. Kim. Plagiarism detection using the Levenshtein distance and Smith-Waterman algorithm. In *Innovative Computing Information and Control*, page 569, 2008.
10. R. A. Wagner and M. J. Fischer. The string-to-string correction problem. *J. ACM*, 21(1):168-173, 1974.