THE CONSTRUCTION OF INDONESIAN-ENGLISH CROSS LANGUAGE PLAGIARISM DETECTION SYSTEM USING FINGERPRINTING TECHNIQUE

Zakiy Firdaus Alfikri and Ayu Purwarianti

STEI, Institut Teknologi Bandung, Jl. Ganesha 10, Bandung, 40132, Indonesia

E-mail: zakiy_f_a@yahoo.co.id

Abstract

Cross language plagiarism detection is an important task since it can protect person intellectual property right. Since English is the most popular international language, we proposed an Indonesian-English cross language plagiarism detection to handle such problem in Indonesian-English domain where the suspected plagiarism document is written in Indonesian and the source document is written in English. To minimize translation error, we build the system by translating the Indonesian document into English and then compare the translated document with the English document collection. The detection system consists of preprocess component, heuristic retrieval component, and detailed analysis component. The main technique used in retrieval process is fingerprinting which can extract lexical features from text which is suitable to be used to detect plagiarism done using literal translation method. In this paper, we also propose additional methods to be implemented in heuristic retrieval component to increase the performance of the system: phrase chunking, stop word removal, stemming, and synonym selection. We evaluated system's performance and the effects of additional methods to system's performance, provided several data test sets which represents a plagiarism type. From the experiments, we concluded that the system works on 83.33% of test cases. We also concluded that mainly all additional methods except the phrase chunking have good effects in enhancing the system accuracy.

Keywords: *plagiarism, detection system, Indonesian-English cross language, fingerprinting, phrase chunking*

Abstrak

Deteksi plagiarisme lintas bahasa merupakan hal yang penting untuk melindungi hak kekayaan intelektual. Bahasa Inggris adalah bahasa internasional yang paling populer, karenanya peneliti mengusulkan deteksi plagiarisme lintas bahasa Indonesia-Inggris untuk menangani masalah tersebut di mana domain dokumen yang diduga plagiat ditulis dalam bahasa Indonesia dan dokumen sumber ditulis dalam bahasa Inggris. Untuk meminimalkan kesalahan terjemahan, peneliti membangun sistem dengan menerjemahkan dokumen bahasa Indonesia ke bahasa Inggris dan kemudian membandingkan dokumen yang diterjemahkan dengan koleksi dokumen bahasa Inggris. Sistem pendeteksian ini terdiri dari komponen *preprocess*, komponen pencarian heuristik, dan komponen analisis detail. Teknik utama yang digunakan dalam temu kembali informasi adalah *fingerprinting* yang dapat mengekstrak fitur leksikal dari teks yang cocok digunakan untuk mendeteksi plagiarisme dengan menggunakan metode terjemahan harfiah. Dalam tulisan ini, peneliti juga mengusulkan metode-metode tambahan yang akan diimplementasikan dalam komponen pengambilan heuristik untuk meningkatkan kinerja system seperti *chunking frase*, penghilangan *stop word*, *stemming*, dan pemilihan sinonim. Peneliti mengevaluasi kinerja sistem dan efek dari metode tambahan untuk kinerja sistem, dengan menyediakan sekumpulan skenario tes beberapa data yang merepresentasikan plagiarisme. Dari pengujian diperoleh kesimpulan bahwa sistem bekerja pada 83,33% kasus uji. Peneliti juga menyimpulkan bahwa terutama semua metode tambahan kecuali *chunking frase* memiliki efek yang baik dalam meningkatkan akurasi sistem.

Kata Kunci: *plagiarisme, sistem deteksi, lintas bahasa Indonesia-Inggris, sidik jari, phrase chunking*

1. Introduction

Plagiarism is a form of cheating which is done by taking the writings of others to put in his own without including the source of origin writings [1]. Plagiarism is a form of idea theft which is a person's intellectual property right [2].

One example of plagiarism is cross-language plagiarism. Cross-language plagiarism is plagiarism which is done by taking writing written in some language and then written back in another

language in his own writing [3]. Although written in different languages, if the content semantically the same then it is plagiarism [4].

In Indonesia, the Indonesian documents could be the result of plagiarism that takes sources from English documents. However, there is no Indonesian-English cross language plagiarism detection system that has been built to resolve the issue. Until now there are only Indonesia monolingual plagiarism detection systems [5] [6] [7]. There are several methods that are applied to the detection of plagiarism monolingual language of Indonesia. These methods are the latent semantic analysis method [5], fingerprinting method [6], and N-rouge, rouge-L, and rouge-W [7].

Cross-language plagiarism detection is more difficult to build than monolingual plagiarism detection. Cross-language plagiarism detection requires a translator component that performs cross-language interface translation. The accuracy of the translation components should also be considered and designed in such a way to make optimum detection results.

In this paper, we construct a system that can detect Indonesian-English cross language plagiarism. Based on research conducted Alzahrani et al., [8] cross-language plagiarism detection is more suitable to use extrinsic plagiarism detection approach. System built in this paper will use extrinsic plagiarism detection approach. For cross language plagiarism detection system, Potthast et al. [3], has designed an architecture that has three main processes. These processes are heuristic retrieval, detailed analysis, and knowledge-based post-processing. The system we propose is designed based on the architecture that has been designed by Potthast et al. [3]. In the system built, heuristic retrieval will use fingerprinting methods and detailed analysis will implement the method CL-C3G. There are also some differences between the system we proposed and the system Potthast had designed.

The rest of the paper is organized as follows: Section 2 presents the methods to detect cross language plagiarism which is proposed by other paper. In section 3 we analyze and design the construction of the detection system then shows the experiments and the results of the system performance. Section 4 draws this paper to a conclusion.

2. Methodology

A cross-language plagiarism detection architecture designed by Potthast et al. [3] can be used to perform cross-language plagiarism detection. There are three main components in the architecture. These components are heuristic retrieval, detailed analysis, and knowledge-based post-processing. These components are described as follow [3]: (1) Heuristic Retrieval - Heuristic retrieval is a component which function is to retrieve documents from corpus that are similar to the inputted document. This component needs a machine translator that will do the translation of inputted document. (2) Detailed analysis - Detailed analysis is a component which function is to compare parts of the inputted document with parts from documents which are selected by heuristic retrieval component. Pair of parts that have high similarity are most likely to be the plagiarism part. (3) Knowledge-based Post-processing - Knowledge-based post-processing is a component that filters the results obtained by detailed analysis process. This component separates the real plagiarism and false positive parts in the inputted document.

First, the input document is processed by heuristic retrieval component. Once the heuristic retrieval component obtains documents that most likely to be the sources of plagiarism, the detailed analysis component is run. The detailed analysis component determines which parts of the inputted document are plagiarisms. Then, knowledge-based post-processing component determines whether the suspected parts really are plagiarism or not.

The proposed Indonesian-English cross language plagiarism detection system has also three main components. The system architecture was designed based on Potthast's architecture for cross language plagiarism detection system. But, there are some differences. The proposed system doesn't have any post-processing component.

The module on filtering false positives is done before the heuristic retrieval component. By this, the sentences which obviously are not plagiarism (e.g. citation) are no longer processed by the system. This process is in preprocessing component which is executed before heuristic retrieval component. So, the proposed system has three components: preprocessing, heuristic retrieval, and detailed analysis component.

Such as mentioned in the previous section, there are 3 components in the system, namely preprocessing, heuristic retrieval, and detailed analysis. Each component is described in the following paragraphs. Preprocessing component aims to filter the citing sentence. It uses pattern matching method to search for citation text in each sentence in input document. Pattern matching is performed to search if there are citations clue in sentences. The patterns contain author's name and publication year of the paper

cited. These sentences will not be processed to the next process.

For example there is a sentence like "Cross-language plagiarism is an important direction of plagiarism detection research but is still in its infancy (Potthast et al, 2010)." The pattern matcher could recognize the brackets '[' and ']' or '(' and ')'. The pattern matcher then analyze if there is a number inside the '[ ]' brackets or if there are a number that represent year, string(s) that represent author's name(s), and a comma that divides them inside the '( )' brackets. From the sample sentence, the pattern matcher can conclude that the sentence is a citation.

The candidate sentences are then inputted into heuristic retrieval component which aim to filter the most possible sentences with plagiarism clue. Basically, there are three methods can be employed in the heuristic retrieval component: fingerprinting, information retrieval, and cross language information retrieval. Since the major concern of the construction of the system is to detect plagiarism which is made by using literal translation method, then we employed fingerprinting method in the heuristic retrieval component. Fingerprinting method is suitable for detecting this kind of plagiarism because it performs heuristic search using lexical features in the text [8].

Fingerprint is a description of an object which is usually in form of a set of number or other data that can be used to characterize an object. Thus, the fingerprint can be used as an indicator of similarity or resemblance between documents. The set of numbers of a fingerprint is calculated using hash function. The numbers represent certain parts of the document. The comparison between the numbers of a document's fingerprint with the numbers of another document's fingerprint is the similarity value between the two documents.

The set of numbers on a document's fingerprint represent the document's characteristic literally rather than semantically. So, two documents that literally similar will have high similarity between their fingerprint but two documents that semantically similar but differ literally will have low similarity between their fingerprint.

Plagiarism made using literal translation tends to have similar lexical features. Therefore, fingerprinting method is chosen as heuristic retrieval's method. Plagiarism that is made by using idea adoption is likely to have different lexical features but still have similar semantic features. Information retrieval and cross language information retrieval method are more suitable to be used to detect plagiarism made using idea adoption. Since the focus is to detect plagiarism which is made by using literal translation, fingerprinting method is the most suitable method to be used in the system. Furthermore, fingerprinting method has faster process because of its efficiency and lightness obtained by using Winnowing in its process [9].

The cross language method in the fingerprinting requires machine translator that will translate the input document. Obviously, the translation accuracy of the translation machine must be considered. Accuracy problems, among others, are OOV (Out of Vocabulary) issue and the selection of appropriate synonym from each word translated. These problems are attempted to be solved using additional methods that will be explained on the next section.

To get more clearly, we provide an example of the processes in heuristic retrieval component. First we have a sentence: "Plagiarisme, yang merupakan pemakaian dari karya orang lain tanpa pengakuan, dianggap sebagai masalah terbesar dalam penerbitan, ilmu pengetahuan, dan pendidikan." This sentence is translated using machine translation becoming "Plagiarism, which is unacknowledged use of other's work, is considered as the biggest problem in publishing, science, and education."

The translated sentence then processed using fingerprinting method. Firstly, the spaces and punctuations in translated sentence are removed. The sentence becomes "Plagiarismwhichisunacknowledgeduseofothersworkisconsideredasthebiggestprobleminpublishingsscienceandeducation". It is then divided into a group of 5-grams. The 5-grams of the sentence are "plagi, lagia, agiar, giari, …, ation". Then, each 5-gram is hashed into a string of integers. The hashing result is a group of strings of integers (figure 1).

92754036	93082290	100504212	99272707
94631429	99288454	99171178	96874925
96684001	97620744	96782669	96693503
96819500	106013626	102466481	100528963
99469115	92637194	94725486	102204225
102848561	96357184	95528563	96667477
100494360	94844760	100049501	95473776
95911029	93735390	96816648	106940343
105537377	93823031	96627267	100358072
93827060	99283257	98647123	94657204
94534197	92957878	95460817	94433148
93141749			

Figure 1. The hashing result.

To choose the numbers to become the fingerprint, the fingerprinting method uses winnowing. The group of hashed numbers is

divided into group of 4-window. Then, they become as showed in figure 2.

```
[92754036  93082290  100504212  99272707]
[93082290  100504212  99272707  94631429]
[100504212  99272707  94631429  99288454]
[99272707  94631429  99288454  99171178]
[94631429  99288454  99171178  96874925]
[99288454  99171178  96874925  96684001]
…
[92957878  95460817  94433148  93141749]
```

Figure 2. Group of 4-window.

From each window we choose one or no number by following some rules. The rules are that in each window select the minimum value. If there is more than one number with the minimum value, select the rightmost occurrence. All the selected numbers are saved as the fingerprint. The fingerprint for the sample sentence is as showed in figure 3.

```
[92754036, 93082290, 100504212, 99272707,
94631430, 99288491, 100511614, 92637194,
94725486, 102204225, 102848553, 96356935,
95520851, 95951404, 96684001, 97620744,
99419471, -997818656, 106013202, 102058497,
100494360, 94844760, 100049501, 95473776,
95356042, 93122303, 96299340, 93735390,
96816648, 106940343, 105537377, 93823031,
96627267, 100358072, 93827060, 99283257,
98647123, 94657204, 94534197, 92957878,
95460817, 94433148, 93141749]
```

Figure 3. The fingerprint.

For detailed analysis component, the system uses cross-language character n-gram model with n equals to three (CL-C3G). CL-C3G is used as in Potthast's paper [3]. CL-C3G is chosen because it has good and stable performance compared to the other methods. This selection is made based on an experiment that compares performances of detailed analysis methods done by Potthast et al. [3].

The experiment shows that cross-language alignment-based similarity analysis model (CL-ASA) has the best result in recall parameter for a collection of document, JRC-Acquis, compared to cross-language semantic analysis model (CL-ESA) and CL-C3G. However, for other collection, the Wikipedia test collection, CL-ASA has significantly worst performance in recall parameter than the two other methods. CL-C3G is likely to have good performance for both collections. CL-C3G performs better than the CL-ESA and is more stable than CL-ASA. Therefore CL-C3G method is chosen to be applied in the system.

The system calculates the similarity of the CL-C3G using fingerprinting method for each sentence in the inputted document. This method creates a fingerprint of the grams which is generated by CL-C3G method. The similarity is calculated based on fingerprint similarity. The process of detailed analysis component is performed on the source documents that exceed the threshold specified in heuristic retrieval component.

We provide an illustration of the processes in detailed analysis component. Each sentence in inputted document is translated. We used the translated sentences obtained by heuristic retrieval component. Each sentence then to be fingerprinted but we used 3 instead of 5-gram to represent C3G. No winnowing is used in this process so the numbers in the fingerprint are quite a lot. Then calculating the similarity between fingerprints is able to be more detailed. We calculate similarity between each sentence in inputted document with each sentence in retrieved document. The similarity is calculated as the percentage of amount of numbers in an inputted sentence's fingerprint that have the same value as the numbers in a retrieved sentence's fingerprint. Pairs of sentences that have more-than-threshold-value similarity are concluded as the plagiarism parts.

To improve the performance of the system, there are some additional methods to be implemented into the system. These methods are phrase chunking, synonym analyzing, stemming, and stop word removal. Phrase chunking is the process that separates a set of words into phrases. This method can be used to eliminate words that do not contribute significantly in a text. Phrase chunking can take only the noun phrases from a sentence. Noun phrases have greater chance as the words which have significant role in the text than other words. Getting high similarity result from the correct document is expected from using the phrase chunking method. With this method the system only processes the words that have significant role in the document.

Synonym analyzing is the process of choosing which words best fit the translation with certain rules. This method is intended to improve the accuracy of the translation done by machine translator. This method is intended to improve the performance of machine translator in translating the document by choosing and replacing word with its most suitable synonym. Heuristic used in this method is the word that appears more in the collection of source documents considered to be

more suitable. If the synonym of a word is greater in number in collection than the word itself then that word will be replaced by its synonym.

Stemming is the process of changing the words to its basic form. Basic form isn't mean its basic word. One example of stemming is the conversion of the word 'writing' into 'writ'. This method is also performed to further increase the similarity value calculated by the system. Stemming makes the similarity between two words become known although they have different tenses in text.

Stop word removal is the removal of words that do not have major influence on the accuracy of the heuristic retrieval component. This process is performed to eliminate words that do not have a significant effect on the detection processes. With stop word removal method, the system performs the detection on the words that are considered to have more significant role in determining the presence of plagiarism. Stop word removal is also performed to decrease the similarity value between documents that have a low similarity. Without stop word removal, different documents may have a high similarity value because of the presence of stop words.

In general, the main components of Indonesian-English cross language plagiarism detection system are preprocess component, machine translation component, heuristic retrieval component, and detailed analysis component. Preprocess component is the component that performs the analysis that decide whether the sentences in the inputted document are citations or not. The citation sentences will not be processed further by the system.

Machine translation component is the component that plays role in translating the inputted document into English. This component uses Google Translate (translate.google.com) as the machine translator. In this translation process, there is additional method implemented to overcome the problems of OOV and synonym selection. The method is synonym analyzing. This additional method uses Java API for WordNet Searching (JAWS) and WordNet 2.1.

Heuristic retrieval component is the component that performs similarity analysis to find documents that have high similarity with the inputted document from corpus. Heuristic retrieval component uses fingerprinting to analyze the similarity between documents. In this component, there are some additional methods implemented: phrase chunking, stop word removal, and stemming. Phrase chunking method uses library from Stanford Parser [10]. Stop word removal uses a list of stop words that is on RCV1 [11]. Stemming uses Porter Stemming library [12].

Detailed analysis component is a component that searches which parts of document are the results of plagiarism based on the selected source documents obtained by the heuristic retrieval component. This component uses fingerprints and CL-C3G to search similarities in the sentences level of a document.

Each component interacts with others, creating system architecture. The architecture of the system built shown in figure 4.

3. Results and Analysis

To know the system performance is obtained by analyzing the level of correctness of the system results. The experiment is executed on a group of document that are the result of plagiarism from some documents. The plagiarism documents are made by using the literal translation. This experiment is done by setting n from n-gram to 5, the length of the window to 4, and the threshold of the retrieval to 40%. Experiment is done without using any additional methods.

This experiment uses a corpus that contains 10 pieces of document in English. Each document in corpus has topic related to NLP and text processing. Fingerprint collection used in experiment is the collection of fingerprints generated from the documents in the corpus.

The data used in the experiment is divided into four big groups of test cases. The following are the test cases: (i) Test case 1 is a collection of documents in which entire text of each document is the result of plagiarism from a source documents. (ii) Test case 2 is a collection of documents in which only some sentences in each document are the result of plagiarism from a source document. (iii) Test case 3 is a collection of documents in which entire text of each document is the result of plagiarism from some source documents. (iv) Test case 4 is a collection of documents in which entire text of each document is the result of plagiarism from a source document which has similarity with another document in corpus.

Experiment results are shown in table I. From the experiment performed on test case 1, it is known that the system produces correct results for each inputted document. From the experiment performed on test case 2, the system does not always give correct results. From the experiment performed on test case 3, it is known that the system produces correct results for each inputted document. And from the experiment performed on test case 4, it is known that the system produces correct results for each inputted document.

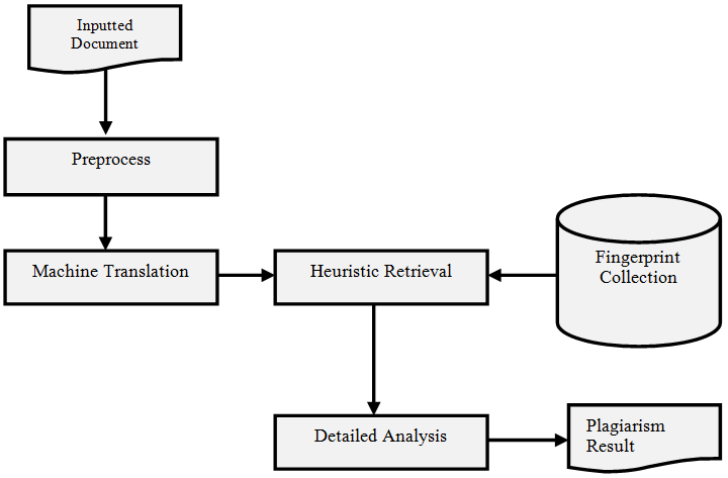

Figure 4. The architecture of the detection system.

From the experiment done, conclusion can be obtained. The conclusion is that the system worked well and can give results as expected. This conclusion is obtained from the results obtained in test case 1, 3, and 4. For test case 2 in which inputted documents only have some sentences as the result of plagiarism, the system gives unsatisfactory results. System gives the expected results if there are substantial portions of the plagiarism, as in the document 3 in test case 2. It can be concluded that the system worked well and can give results as expected for cases of whole plagiarism documents with a source of plagiarism, plagiarism documents with more than one sources, and documents with a plagiarism source that have similarities with other documents in the corpus.

TABLE I
THE EXPERIMENT RESULT

Test Case	Inputted Document	Detection Result	Result Correct-ness	Similarity Value
1	1	Found: Doc 5	Correct	76.04%
	2	Found: Doc 6	Correct	73.54%
	3	Found: Doc 8	Correct	71.60%
2	1	Found: Doc 2	False	47.28%
	2	Found: Doc 2	False	46.29%
	3	Found: Doc 9	Correct	48.16%
3	1	Found: Doc 3	Correct	56.61%
	2	Found: Doc 2	Correct	77.47%
	3	Found: Doc 4	Correct	67.47%
4	1	Found: Doc 2	Correct	78.88%
	2	Found: Doc 3	Correct	78.68%
	3	Found: Doc 10	Correct	72.39%

In this section, we show the results obtained by the system using additional methods. We also show the effects of the additional method on the result obtained by the system. As for the experimental data, we employed the same corpus as in the experiments explained in the previous section. And for the test cases, one document for each case mentioned in the previous section is used.

The data used in the experiment is divided into four big groups of test cases. The following are the test cases: (i) Test case of phrase chunking.(ii) Test case of synonym analyzing. (iii) Test case of stop word removal. (iv) Test case of stemming.

The experiment is done by testing additional methods implemented in the system for each test document. The experiment results are shown in table II. From the experiment can be shown that the additional method that has good effect to the system performance is phrase chunking. Stop word removal and stemming have effects which are not very significant in influencing the results obtained by the system.

Phrase chunking method can significantly improve the system performance by increasing the similarity value of the documents in corpus. Synonym analyzing method can increase the detection sensitivity by decreasing the similarity value. Stop word removal and stemming are not so influential in the system which uses fingerprinting. These methods do not result in significant influences on the system performance.

Fingerprinting processes text which are firstly formed into a collection of 5-gram. This makes stop words which are usually in short sizes to be incorporated with part of another words. This indirectly reduces the likelihood of similarity that could occur by the presence of stop words in the text. Stop word removal methods do not have much effect on the results obtained since the system eliminates the stop word function which is performed by the fingerprinting process. However, by the presence of this method, the system will work faster especially in calculating the

fingerprint input from documents because the amount of text to be processed is reduced.

Stemming method is not so influential in increasing or giving change in the results obtained by the system. Stemming affects on lexical features that exist in the inputted document. Apparently this is not very influential. This can be caused by the fact that fingerprinting method divides the text into the form of 5-gram which has been effective enough to get the similarity of text in small portions so that the stemming process that will change the shape of a word into its basic form does not affect significantly.

TABLE II
THE EXPERIMENT RESULTS

Method	Test Document	Result Using Standard Method (Similarity Value)	Result Using Additional Method (Similarity Value)
Phrase chunking	1	Found: Doc 5 (76.04%)	Found: Doc 5 (85.97%)
	2	Found: Doc 2 (47.28%)	Found: Doc 9 (62.02%)
	3	Found: Doc 2 (77.47%)	Found: Doc 2 (82.32%)
	4	Found: Doc 3 (78.68%)	Found: Doc 3 (83.68%)
Synonym analy-zing	1	Found: Doc 5 (76.04%)	Found: Doc 5 (60.48%)
	2	Found: Doc 2 (47.28%)	Found: Doc 2 (44.64%)
	3	Found: Doc 2 (77.47%)	Found: Doc 2 (67.80%)
	4	Found: Doc 3 (78.68%)	Found: Doc 3 (62.82%)
Stop word removal	1	Found: Doc 5 (76.04%)	Found: Doc 5 (75.67%)
	2	Found: Doc 2 (47.28%)	Found: Doc 2 (47.28%)
	3	Found: Doc 2 (77.47%)	Found: Doc 2 (77.52%)
	4	Found: Doc 3 (78.68%)	Found: Doc 3 (78.86%)
Stemm-ing	1	Found: Doc 5 (76.04%)	Found: Doc 5 (68.16%)
	2	Found: Doc 2 (47.28%)	Found: Doc 2 (45.68%)
	3	Found: Doc 2 (77.47%)	Found: Doc 2 (77.44%)
	4	Found: Doc 3 (78.68%)	Found: Doc 3 (77.80%)

4. Conclusion

The conclusion obtained from the work of this paper is that the use of fingerprinting methods in the process of heuristic retrieval is suitable to implement in system that detect plagiarism which is created using a literal translation techniques. CL-C3G methods and fingerprinting are also suitable to be used in the detailed analysis component of the system. The performance and accuracy of the system could be improved by using some appropriate method. Phrase chunking method has the effect of increasing the value of similarity for each collection of tested documents. Synonym analyzing method tends to lower the value of the similarity that makes the detection of similarities become more sensitive. Stop word removal and stemming methods are not so significant in improving the results obtained by the system that uses fingerprinting methods as its heuristic retrieval component.

For further works, it is necessary to compare the performance of Indonesian-English cross language plagiarism detection system that use fingerprinting as constructed in this paper with systems that use information retrieval and cross-language information retrieval methods. Then, the synonym analyzing method should be further developed until it is completely suitable to be used in cross-language plagiarism detection system and have significant influence in obtaining the correct results. It should be also carried out researches to find other methods that may have significant influences in making the detection system to obtain optimal results.

References

[1] A. Barron-Cedeno, P. Rosso, D. Pinto, & A. Juan, "On Cross Lingual Plagiarism Analysis Using Statistical Model" *In Proceeding PAN*, pp. 21-24, 2008.

[2] H. Maurer, F. Kappe, & B. Zaka, "Plagiarism – A Survey," *Journal of Universal Computer Science*, vol. 12, no. 8, pp. 1050-1084. 2006.

[3] M. Potthast, A. Barron-Cedeno, B. Stein, & P. Rosso, "Cross-Language Plagiarism Detection," *Lang Resources & Evaluation,* vol. 45, pp. 45-62. 2010.

[4] A. Barron-Cedeno & P. Rosso, "Monolingual and Crosslingual Plagiarism Detection" *In Proceeding Competition SEPLN09 In: III Jornadas PLN-TIMM*, pp. 29-32, 2009.

[5] A. Ardiansyah, "Pengembangan Aplikasi Pendeteksi Plagiarisme Menggunakan Metode Latent Semantic Analysis (LSA)", 2011.

[6] P. Kusmawan, U. Yuhana, & D. Purwitasari. "Aplikasi Pendeteksi Penjiplakan pada File Teks dengan Algoritma Winnowing", B.S Thesis, Teknik Informatika, ITS, Indonesia, 2009.

[7] F. Mahathir, "Sistem Pendeteksi Plagiat pada Dokumen Teks Berbahasa Indonesia Menggunakan Metode Rouge-N, Rouge-L, dan Rouge-W", 2011.

[8] S. Alzahrani, N. Salim, & A. Abraham, "Understanding Plagiarism Linguistic Patterns, Textual Features and Detection Methods," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,* vol.42, pp. 133-149, 2011.

[9] S. Schleimer, D. Wilkerson, & A. Aiken, "Winnowing Local Algorithms for Document Fingerprinting" *In SIGMOD 03 Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pp. 76-85, 2003.

[10] D. Klein & C. Manning, "Accurate Unlexicalized Parsing" *In Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pp. 423-430, 2003.

[11] D. Lewis, Y. Yang, T. Rose, & F. Li, "RCV1: A New Benchmark Collection for Text Categorization Research," *Journal of Machine Learning Research,* vol. 5, pp. 361-397. 2004.

[12] M. Porter, "An Algorithm for Suffix Stripping," *Program*, vol. 14, pp 130-137. 1980.