TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol. 12, No. 11, November 2014, pp. 7884 ~ 7894
DOI: 10.11591/telkomnika.v12i11.6652

Detailed Analysis of Extrinsic Plagiarism Detection System Using Machine Learning Approach (Naive Bayes and SVM)

Zakiy Firdaus Alfikri*[1], Ayu Purwarianti[2]
Institut Teknologi Bandung. Jl. Ganeca 10, Bandung, Indonesia
*Corresponding author, e-mail: zakiy_f_a@yahoo.co.id[1], ayu@stei.itb.ac.id[2]

Abstract

In this report we proposed a detailed analysis method of plagiarism detection system using machine learning approach. We used Naive Bayes and Support Vector Machine (SVM) as learning algorithms. Learning features used in the method are words similarity, fingerprints similarity, latent semantic analysis (LSA) similarity, and word pair. Those features are adapted from some state-of-the-art methods in detailed analysis of a plagiarism detection system. The purpose in selecting those features is to retrieve information from the state-of-the-art detailed analysis methods (words similarity, fingerprinting, and LSA) in order to integrate the strength of each method in detecting plagiarism. Several experiments were conducted to test the performance of the proposed method in detecting many cases of plagiarism. The experiments used 70 data test. The data test contains cases of literal plagiarism, partial literal plagiarism, paraphrased plagiarism, plagiarism with changed sentence structure, and translated plagiarism. The data test also contains cases of non-plagiarism of different topics and non-plagiarism of the same topic. The results obtained in experiments using SVM showed an average accuracy of 92.86% (reaching 95.71% without using words similarity feature). While the result obtained using Naive Bayes showed an average accuracy of 54.29% (reaching 84.29% without using the word pair features). Using SVM algorithm showed better results because it is naturally suitable in classifying problems that have two classes and it is better than Naive Bayes in resolving high-dimensional problems (which have a lot of features). The proposed method (using SVM) has an average of high accuracy for each of the tested cases of plagiarism. It proves that the proposed method (using SVM) is able to integrate the information in detecting plagiarism from state-of-the-art detailed analysis method (words similarity, fingerprinting, and LSA) to obtain a more accurate detection results.

Keywords: *detailed analysis, plagiarism detection, machine learning approach, learning algorithm, learning feature*

1. Introduction

Plagiarism is a form of cheating which is done by taking the writings of others and then put in his own without any credit given to the origin [1]. Likewise, according to IEEE, plagiarism is reuse of ideas, processes, results, or words of another person without giving information about the original author and source explicitly [2]. Plagiarism is theft of idea which is a person's intellectual property right [3].

There are some previous works in Indonesian monolingual (single language) plagiarism detection studied by [4-7]. In addition, there is Indonesian-English cross-language plagiarism detection studied by [8]. Plagiarism detection needs to be designed so it does not depend on whether the plagiarism is monolingual or cross lingual.

There are two approaches to detect plagiarism. They are extrinsic and intrinsic plagiarism detection approaches [9]. Plagiarism detection in this study is using extrinsic plagiarism detection approach because it is expected to be able to detect the presence of intelligence plagiarism that is performed using the idea adoption and translation.

General architectural of plagiarism detection system consists of three main stages, namely heuristic retrieval, detailed analysis, and knowledge-based post-processing [10]. Heuristic retrieval is the process of retrieving documents from corpus that are likely to be the source of plagiarism. We named them candidate documents. Detailed analysis is the process of searching similarities between the input document and the candidate documents at a more

Received September 8, 2013; Revised October 2, 2014; Accepted October 10, 2014

detailed level (sentence or paragraph). Knowledge-based post processing is the process of filtering false positives that might be produced from previous processes.

There are several state-of-the-art methods that can be used to perform detailed analysis. Methods such as fingerprinting and latent semantic analysis (LSA) have been used to perform detailed analysis in plagiarism detection.

Fingerprinting methods tends to have a high performance for cases of literal plagiarism, while the LSA method tends to have good performance for cases of intelligence plagiarism such as plagiarism made using paraphrasing. Fingerprinting method has weaknesses in detecting cases of plagiarism made using paraphrasing. As for the LSA method cannot detect the sentence pairs that are not plagiarism but still stand in one topic.

To obtain information in detecting plagiarism from fingerprinting and LSA methods and to get the combined strength of each state-of-the-art method, we proposed a method of performing detailed analysis based on machine learning approach. Machine learning is a method of improving performance in line with the experiences made on a particular task. Machine learning has shown to have a high utility value for a variety of application domains [11]. Quality of the machine learning depends on the selection of training experience (training data feature), the target function and its representation, and its learning algorithms.

In this study we used two learning algorithms. They are Naive Bayes and Support Vector Machine (SVM) algorithms. We used SVM learning algorithms because it has excellent performance in text classification problems that can be seen in the experiment results of [12] and [13]. SVM algorithm is also suitable for detecting plagiarism problem because naturally SVM is suitable for classifying problems that have two classes, as had been analyzed by [14]. Then Naive Bayes algorithm is chosen because it is suitable to be used as comparative baseline in text classification problems [15] as can be seen in experiments of [12] and [13].

2. Detailed Analysis of Plagiarism Detection System

2.1. Similarity Measurement Methods

There are many methods that can be used to compare the similarity between sentences. Some of them are fingerprinting, vector space model (VSM), and latent semantic analysis (LSA).

In the fingerprinting method, the amount of similar fingerprints is used as similarity indicator between sentences. Fingerprint is a statement that can characterize an object [16]. Fingerprint of a sentence is in the form of integer values that are calculated using the hash function [17]. Measurement of similarity between sentences is calculated by comparing the similarity portion between a sentence's fingerprint and another sentence's fingerprint.

In the vector space model method, the sentences are represented in vector form that is based on the weight of the words in the sentence [18]. Similarity measurement between sentences is calculated using cosine similarity function between the vectors of the sentences.

In the latent semantic analysis method, a matrix is formed that represents the term-sentence matrix. This matrix is then split using singular value decomposition (SVD) to become three matrices. They are matrix U representing terms, matrix V^T representing the sentences, and the matrix S which is a diagonal matrix of singular value [19]. Similarity measurement is calculated using the cosine similarity function between the vectors formed by the matrices.

2.2. Similarity Measurement Method Using Machine Learning Approach

Machine learning is a branch of artificial intelligence. Machine learning can construct a learning model from available training data. The goal of using machine learning is to improve performance (based on a particular performance measure) to a task based on a learning model constructed from training data (experiences) [11].

Machine learning can solve many kind of problems. Machine learning can be used to decide if two sentences are plagiarism or not from the information contained in them. So, machine learning can be used as an approach in detailed analysis method. The use of machine learning in detailed analysis requires training data in the form of collection of sentence pairs and their label (plagiarism or not). From the collection of the training data, machine learning will extract the appropriate features such as words similarity, fingerprint similarity, LSA similarity, and so on.

If a lot of features are used, we can do feature selection. Feature selection is performed to get the number of features that are not too big but still be able to represent the information required. Several feature selection methods are threshold frequency selection and mutual information ranked selection.

Frequency threshold is the minimum frequency limit possessed by a feature to be able to qualify for selection. Mutual information is a dependency value of a feature in deciding the value of the class. It will rank the features based on mutual information values. Top *n*-features are selected from the resulting ranking.

A learning model can be built from the features obtained by the previous process. Using the learning model, we can classify whether a pair of sentence is a case of plagiarism or not. By using appropriate features, machine learning methods can be used to see plagiarism on two different document languages.

There are many kinds of learning algorithms that can be used in machine learning. Some of them are Naive Bayes and Support Vector Machine.

Naive Bayes is a learning algorithm that represents each instance (data) as the conjunction of its attributes values [11]. Naive Bayes can only classify the data that has limited values of target function (classes); it means the class' value cannot be continuous. Naive Bayes classifies the data by selecting the class that has the highest chance value, assuming that each attributes independent to each other.

Naive Bayes is based on Bayes rule. It assumes the attributes a_1 ... a_n are all conditionally independent of one another, given V [11]. It can be described by:

$$V_{class} = \arg\max_{V_j \in V} P(a_1, a_2, \ldots, a_n | V_j) P(V_j) \quad (1)$$

Support Vector Machine (SVM) is learning algorithm that analyze data and recognize patterns [14]. SVM constructs a hyperplane or set of hyperplanes in a high-dimensional space, which can be used for classification. It can make a hyper plane that can separate two classes [14]. This algorithm analyzes the data, looks for a pattern, and then creates hyper plane separator to divide the data based on each class. SVM models the existing data into points in a space. The location of each point depends on the value of the features used.

3. Analysis of Plagiarism Detection Detailed Analysis

3.1. Plagiarism Cases

There are several cases of plagiarism that may occur. The followings are some cases of plagiarism.

a) Literal plagiarism, which is plagiarism performed by copying the source text directly without modification.
b) Partial literal plagiarism, which is plagiarism performed by copying small portion of source text.
c) Paraphrased plagiarism, which is plagiarism performed by paraphrasing the copied source text.
d) Plagiarism with changed sentence structure, which is plagiarism performed by changing the structure of the copied sentence.
e) Translated plagiarism, which is plagiarism performed by translating the copied text.

There are also cases that are not plagiarism. The followings are some cases that are not plagiarism.

a) Non-plagiarism with different topic, which is the case of the sentence and the other are not plagiarism and their contents are in different topic.
b) Non-plagiarism but in one topic, which is the case of the sentence and the other are not plagiarism but their contents have the same topic.

3.2. Feature Selection

To create model and to perform classification, we need to define which learning features to be used. The selection of features is based on each feature's influence on the class value (plagiarism or not). The followings are the features to be used in proposed method. To explain more about the features, as example we used sentence "POS tag can be obtained

using HMM technique" as to-be-detected sentence and sentence "POS tag can be obtained using HMM technique" as source sentence.

a) Word pairs

Word pairs are chosen to be the features in creating model and performing classification because word pairs that exist between two sentences can determine whether they are plagiarism or not. Word pairs features are expected to give information of different words that have similar context.

We used stop word removal and stemming on words. We listed all possible word pairs from the training data and filtered them using mutual infromation ranking and frequency threshold to generate word pairs. For example, the generated word pairs are obtain_technique, obtain_obtain, technique_ technique, technique_text, algorithm_model, and algorithm_technique. So the sample sentence has attribute value of 1for obtain_technique, obtain_obtain, and technique_technique. And it has attribute value of 0 for technique_text, algorithm_model, and algorithm_ technique.

b) Words similarity

Words similarity is chosen to be one of the features because plagiarism sentence tends to have a high level similarity with the source sentence. But to decide plagiarism or not by looking only at the words similarities is not guaranteed to be always right. For example, the value of this attribute for sample sentence is 1, because to-be-detected and source sentences have 100% words similarity.

c) Fingerprint similarity

Fingerprint similarity is chosen because it provides information of similarity between sentences in a more detailed level, which is in the level of *n*-gram structure. The similarity is the percentage of similar fingerprint between sentences. Fingerprint gives a more detailed similarity value (the *n*-gram) rather than words similarity. After processing the sample sentences, the values of fingerprint for both sentences are the same. So, the value of this attribute is 1.

d) LSA similarity

LSA similarity is chosen because it provides information in the form of conceptual similarity of context (semantics) between the two sentences. Conceptual similarity of context has a great influence in determining plagiarism between sentences. We calculate the cosine similarity between to-be-detected sentence semantic vector and source sentence semantic vector. The result is 1, so the value of this attribute is 1.

3.3. Preprocess Supplementary Components

In preprocess, we used supplementary components such as stop word removal component and stemming component. Stop word removal component will delete stop words that exist on the sentences. This component is utilized to reduce the amount of features generated and to avoid the generation of low-influential features. Stemming is used to get a stem form of every word that exists in the sentence. This component is used to reduce variation of features that have the same context and to generate more high-influential features.

3.4. Architectures

Detailed analysis architecture is divided into two main parts. The two main parts are model creation and classification.

In model creation, firstly preprocessing the training data is performed. Then, it generates word pairs features from the training data. After that, it extracts the values of each feature from the training data. And finally, it creates a learning model that can be used for classification.

Classification preprocesses the two inputs (text to be detected and the candidate text), extracts the values of each feature, and classifies the inputs as plagiarism or not plagiarism using the learning model. Classification architecture can be seen in Figure 1.

3.5. How The System Works

The proposed detailed analysis method requires model creation and classification. For model creation, the first thing conducted is the preprocessing of training data. The preprocessing takes the value of the sentences detected, the source-candidate sentence, and the label of each data. In this processed, stop word removal and stemming are also performed. Next, on the feature extraction process, firstly the generation of word pairs features is

performed. Word pairs features generated can be selected using the mutual information ranking and frequency threshold to reduce the features and to get only high-influential features are to be used.

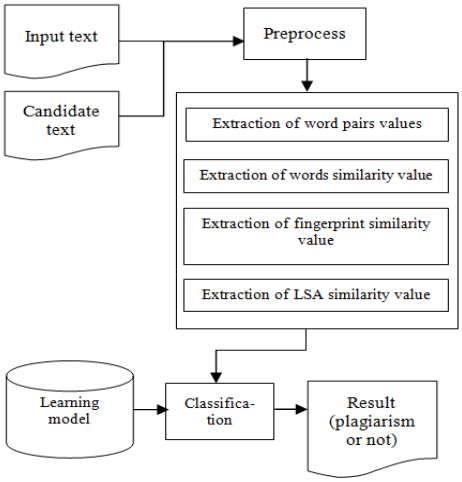

Figure 1. Classification Architecture

Then, the extraction of each feature value is executed for each of the existing data in the training data. Value of the word pair feature is the number of occurrences of the word pair in each data. The words similarity value is the percentage of words similarity between two sentences in each data. Fingerprint similarity value is the percentage of similar fingerprint between two sentences in each data. Then LSA similarity value can be calculated using cosine similarity function between sentences-context vectors of the two sentences in each data. The results of this process are data with their features' value.

After the feature extraction process, the model creation process is performed using the chosen learning algorithm. The learning algorithm will build a model to fit the training data.

For classification, according to Figure 1, the first thing carried out is the preprocessing of the inputs by taking the value of to-be-detected sentence and source-candidate sentence. In preprocessing, stop word removal and stemming processes are also performed. Next, the process of feature extraction, extraction of each feature value for each of the existing data on the training data is performed. The results of this process are data with their features' value.

Finally, the classification of the inputs is performed. It uses the input data as instances that have features' values that is calculated in feature extraction process. Learning algorithm performs classification using the learning model generated by modeling subsystem. The result of the classification is the decision whether the input data is plagiarism or not.

4. Experiments

4.1. Model Creation

In this experiment, model creation processes were performed. It used 80 training data that are plagiarism cases and 80 training data that are not cases of plagiarism. It used all of the features. The value of frequency threshold used was 1 and the value of mutual information used was 10000-first-ranked.

There were two experiments performed, using Naive Bayes and using Support Vector Machine (SVM). The accuracy of each model was calculated using 10-fold cross-validation

4.2. Detecting Plagiarism Cases

In experiments of detecting plagiarism cases, the test cases were created and their data illustrate possible cases of plagiarism. Detecting or classifying existing test data for each test case were also performed. The result also to be compared to the results obtained with the state-of-the-art detailed analysis methods such as word similarity method, fingerprinting, and LSA.

4.2.1. Testing data

The data used for testing consisted of 70 data that were divided into six test cases. The followings are the test cases.

a) Literal plagiarism sentence pairs
b) Partial literal plagiarism sentence pairs
c) Paraphrased plagiarism sentence pairs
d) Changed structure plagiarism sentence pairs
e) Translated plagiarism sentence pairs
f) Non-plagiarism sentence pairs which are different in topic
g) Non-plagiarism sentence pairs which have same topic

4.2.2. Experiment Scenario

In this experiment, the accuracy of the classifier was calculated. The experiment used test cases as the input of the classification processes. The accuracy was obtained by calculating the percentage of correctly classified instances. The experiment used all of the features. The value of frequency threshold used was 1 and the value of mutual information used was 10000-first-ranked.

4.3. Testing The Effect of Frequency Threshold and Mutual Information

This experiment was conducted to see the effect of changing the values of frequency threshold and mutual information parameters that play a role in the selection for word pairs features. Experiment performed by creating model and classifying test data.

The experiment used all of the features. The values of frequency threshold tested were 1, 2, 3, and 4 and the values of mutual information used were 10000, 5000, 7000, and 10000-first-ranked.

4.4. Testing Each Feature Influence

This experiment was conducted to see the effect of each feature and also to see the performance of each learning algorithm used. The value of frequency threshold used was 1 and the value of mutual information used was 10000-first-ranked. Experiment performed by creating model and classifying test data. The experiment was divided into five experiment cases, which were:

a) Using all features
b) Using all features except word pairs features
c) Using all features except LSA similarity feature
d) Using all features except fingerprint similarity feature
e) Using all features except words similarity feature

5. Result

5.1. Result and Analysis of Model Creation Experiment

The accuracy result of model creation experiment can be seen in Table 1. Judging from the accuracy evaluation of model creation, the created learning model has a pretty good performance. Model created using SVM has an accuracy rate of 84.375%. It is because of the features selected to be used are relevant and appropriate in resolving problems of detecting plagiarism in detailed analysis.

For the learning algorithm, it can be seen that the SVM has better performance than Naive Bayes. SVM has a better performance due to the characteristics of its learning that fits high dimensional data. Additionally, because there are only two classes (plagiarism or not) SVM naturally only needs to create a hyper plane that separates the two parts of the class.

Table 1. The Accuracy of Model Creation

Learning algorithm	Accuracy
Naive Bayes	76.25 %
SVM	84.375

5.2. Result and Analysis of Detecting Plagiarism Cases Experiment

The experiment results are presented in Table 2. The data in the table show the accuracy of each model in classifying the test data were used to classify the input.

The experiment results show an average accuracy of 92.86% for the SVM models and an average accuracy of 54.29% for Naive Bayes models. SVM model could classify the types of literal plagiarism, partial literal plagiarism, and translated plagiarism with accuracy rate of 100%. For other cases of plagiarism, such as plagiarism using paraphrasing and changing in sentence structure, the accuracy of the SVM is still very high (90% accuracy). In the case of non-plagiarism with different topic, the accuracy of the resulting model of SVM is also quite high, 90% accuracy. Then for non-plagiarism within the same topic, SVM only reach 80% accuracy.

It is shown that SVM works better than Naive Bayes in almost all cases of plagiarism. SVM has better accuracy in detecting literal plagiarism, partial literal plagiarism, paraphrased plagiarism, and changed structure plagiarism. SVM is also better in detecting non-plagiarism cases. Naive Bayes has the same performance in detecting translated plagiarism.

Table 2. The Accuracy of Detecting Plagiarism Cases

Test case	Naive Bayes Accuracy	SVM Accuracy
1. Literal	50 %	100 %
2. Partial literal	40 %	100 %
3. Paraphrased	50 %	90 %
4. Changed structured	50 %	90 %
5. Translated	100 %	100 %
6. Non-plagiarism with different topic	30 %	90 %
7. Non-plagiarism in the same topic	60 %	80 %
Mean:	54.29 %	92.86 %

Generated Naive Bayes models tend to exhibit poor accuracy. Almost all cases of plagiarism accuracy are not more than 50%, except for case of translated plagiarism that reaches 100%. So are the cases of non-plagiarism, the average accuracy obtained is only 45%. Judging from the results obtained, Naive Bayes models are less suitable for the use as the classifier solution in detecting plagiarism. Reason which make the results have poor accuracy is the number of features used (word pairs features consist of about 1000 feature) in the fact that the training data used is only about 160 data, so Naive Bayes can't form a statistical/probabilistic model that is relevant to be used for the test data.

Table 3. Comparison between State-of-the-art Method and the Proposed Method (using SVM)

Test case	Words similarity	*Finger-printing*	LSA	SVM
1. Literal	100 %	100 %	100 %	100 %
2. Partial literal	90 %	90 %	100 %	100 %
3. Paraphrased	50 %	40 %	90 %	90 %
4. Changed structured	100 %	80 %	100 %	90 %
5. Translated	100 %	80 %	100 %	100 %
6. Non-plagiarism with different topic	100 %	100 %	100 %	90 %
7. Non-plagiarism in the same topic	100 %	100 %	0 %	80 %
Mean:	91.42 %	84.29 %	84.29 %	92.86 %

This experiment also conducted a comparison experiment of the results obtained using SVM models with another state-of-the art detailed analysis method. Comparison of the results can be seen in Table 3.

Can be seen in the average level of accuracy using the proposed SVM method is on the top of the other methods. Viewed from each case in general, the proposed SVM method accuracy of almost all cases is above average. Only in the case of structure changed plagiarism and non-plagiarism, the accuracy is slightly below average.

In the case of literal plagiarism all methods have a very high accuracy. In the cases of partial literal plagiarism and changed structure, the word similarity and fingerprinting method has worse accuracy than other methods. In the case of translated plagiarism, all methods have fairly good accuracy. However, other methods need a machine translation while the proposed method doesn't need any machine translation. And in the case of non-plagiarism within the same topic, LSA method has an accuracy that is far below average. It can be seen that the proposed method combines the strength of the other methods.

5.3. Result and Analysis of Testing The Effect of Frequency Threshold and Mutual Information

The result of testing the influence of frequency threshold and mutual information parameter can be found at Table 4. It can be seen from the evaluation accuracy that the average accuracy of Naive Bayes model is around 73% and the average accuracy of SVM model is around 83%. The most optimal result obtained on frequency threshold value of 1 with the value of n on mutual information is 10000 and on frequency threshold value of 2 regardless of the value of n on mutual information. This is because for 160 training data the amount of most-influential word pairs is only around 10000 pairs of word with value of its frequency boundary is only 1.

Table 4. The Accuracy of each Model for each Parameter

	Frequency threshold: 1	Frequency threshold: 2	Frequency threshold: 3	Frequency threshold: 4
n mutual information: 1000	NB: 74.375 % SVM: 81.875 %	NB: 75 % SVM: 85.625 %	NB: 73.125 % SVM: 84.375 %	NB: 71.875 % SVM: 78.75 %
n mutual information: 5000	NB: 74.375 % SVM: 81.875 %	NB: 75 % SVM: 85.625 %	NB: 73.125 % SVM: 84.375 %	NB: 71.875 % SVM: 78.75 %
n mutual information: 7000	NB: 76.25 % SVM: 83.125 %	NB: 75 % SVM: 85.625 %	NB: 73.125 % SVM: 84.375 %	NB: 71.875 % SVM: 78.75 %
n mutual information: 10000	NB: 76.25 % SVM: 84.375 %	NB: 75 % SVM: 85.625 %	NB: 73.125 % SVM: 84.375 %	NB: 71.875 % SVM: 78.75 %

Table 5. The Accuracy of each Model in Classifying Test Data for each Parameter

	Frequency threshold: 1	Frequency threshold: 2	Frequency threshold: 3	Frequency threshold: 4
n mutual information: 1000	NB: 54.29 % SVM: 92.86 %	NB: 54.29 % SVM: 80.00 %	NB: 54.29 % SVM: 85.71 %	NB: 54.29% SVM: 84.29 %
n mutual information: 5000	NB: 54.29 % SVM: 92.86 %	NB: 54.29 % SVM: 80.00 %	NB: 54.29 % SVM: 85.71 %	NB: 54.29% SVM: 84.29 %
n mutual information: 7000	NB: 54.29 % SVM: 92.86 %	NB: 54.29 % SVM: 80.00 %	NB: 54.29 % SVM: 85.71 %	NB: 54.29% SVM: 84.29 %
n mutual information: 10000	NB: 54.29 % SVM: 92.86 %	NB: 54.29 % SVM: 80.00 %	NB: 54.29 % SVM: 85.71 %	NB: 54.29% SVM: 84.29 %

Then for testing classification of test data, as can be seen in Table 5, the average accuracy of Naive Bayes classification is approximately 54% and the average accuracy of SVM classification is approximately 85%. The most optimal results obtained in the frequency threshold of 1 regardless of the value of n on the mutual information.

If the classification test results are analyzed along with the model creation result, the optimal result is obtained at threshold frequency parameter by 1 and n on mutual information by 10000. As already discussed, this is because for 160 training data the amount of most-influential word pairs is only around 10000 pairs of word with value of its frequency boundary is only 1.

5.4. Result and Analysis of Testing Each Feature Influence

The result of testing the effect of each feature influence to the accuracy of model creation can be seen in Table 6. It can be seen that in the case of 2 to 5 the accuracy of SVM are reduced compared to accuracy in case 1. This proves all the features are influential in improving the accuracy of SVM. Then for Naive Bayes, in the case of 2 the accuracy is increased. This show word pair features are less suitable for Naive Bayes. While the features other than the pairs have an effect in improving the accuracy of Naive Bayes.

Table 6. The Accuracy of each Model in each Experiment Case

	Naive Bayes Accuracy	SVM Accuracy
Case 1	76.25 %	84.375 %
Case 2	77.5 %	78.75 %
Case 3	75 %	80.625 %
Case 4	71.25 %	77.5 %
Case 5	71.875 %	80.625 %

In addition to testing model creation, the testing of test data classification is also performed. Each test case attempted to seven types of test data that has been defined.

For case 1, the results obtained can be seen in Table 2. These results will be used as a comparison for another experiment cases.

For case 2, the average accuracy of Naive Bayes classification increased by about 30%. Accuracy of 100% happens for almost all cases except cases of plagiarism using paraphrasing (that still has a high accuracy, 90%) and translated plagiarism (down to 0% accuracy).

Average accuracy of SVM classification is decreased for about 14.29%. A significant decrease happens in cases of translated plagiarism and non-plagiarism within one topic.

For case 3, the average accuracy of Naive Bayes classification is not different from the average classification accuracy using all features. Average accuracy of SVM classification is decreased for about 12.86%. Significant decrease in accuracy occurred in the case of paraphrased plagiarism and changed structure plagiarism.

For case 4, the average accuracy of Naive Bayes classification is not different from the average classification accuracy using all features. Average accuracy of SVM classification is decreased for about 10%. Significant decrease in the accuracy of the case is not plagiarism in the case of different topics and one topic is not plagiarism.

For case 5, the average accuracy of Naive Bayes classification is not different from the average classification accuracy using all features. Change in the average classification accuracy of SVM is not significant enough, only about 2.85% change. Words similarity feature does not have a significant effect in increasing the accuracy of classifiers plagiarism.

6. Conclusion

The conclusion that can be drawn from this paper is the performance of proposed detailed analysis of plagiarism detection using machine learning approaches is quite high. The results obtained in experiments using SVM showed an average accuracy of 92.86% (reaching 95.71% without using words similarity feature). While the result obtained using Naive Bayes showed an average accuracy of 54.29% (reaching 84.29% without using the word pair features).

Detailed analysis performance varied in each case. At all test cases, proposed detailed analysis performance is quite well, the worst accuracy was 80% for cases of non-plagiarism in similar topic. Compared with the state-of-the-art detailed analysis methods such as word similarity method, fingerprinting method, and LSA method, the proposed method has more advantages in detecting plagiarism and non-plagiarism in almost all cases. Other methods have shortcomings (as seen from the low accuracy) in specific cases. It can be concluded that the proposed method can retrieve information from the other state-of-the-art methods. It can combine the information to obtain the high level of plagiarism detection accuracy.

The features that are most suitable to be used in detailed analysis based on machine learning are fingerprint similarity feature, LSA similarity feature, and word pairs features. And the most suitable learning algorithm to be used is the SVM learning algorithm. Using SVM algorithm showed better results because it is naturally suitable in classifying problems that have two classes and it is better than Naive Bayes in resolving high-dimensional problems (which have a lot of features). Using SVM learning algorithm with fingerprint similarity feature, LSA similarity feature, and word pairs features, the accuracy obtained is in average of 95.71%. This result shows a good performance of the proposed method in detecting extrinsic plagiarism in detailed level.

For further research, experiments in detecting multi-language (other than Indonesian-English) plagiarism may be performed using proposed detailed analysis method in this paper. Training data for related language is required for the experiment. Then, a research may be done to study suitable feature selection method for the word pair features that allow the generation of optimal feature selection to be used exclusively in the plagiarism detection issues.

References

[1] Barron-Cedeno A, Rosso P, Pinto D, Juan A. On Cross Lingual Plagiarism Analysis Using Statistical Model. Prosiding *PAN 2008*. Patras, Yunani. 2008.
[2] IEEE. A Plagiarism FAQ. *http://www.ieee.org/publications_standards/ publications/rights/plagiarism _FAQ.html.* Diakses tanggal 13 Juni 2012.
[3] Maurer H, Kappe F, Zaka B. Plagiarism – A Survey. *Journal of Universal Computer Science.* Austria. 2006*;* 12(8): 1050-1084.
[4] Ardiansyah A. Pengembangan Aplikasi Pendeteksi Plagiarisme Menggunakan Metode Latent Semantic Analysis (LSA). Bandung, Indonesia. 2011.
[5] Kusmawan P, Yuhana U, Purwitasari D. Aplikasi Pendeteksi Penjiplakan pada File Teks dengan Algoritma Winnowing. Indonesia. 2009.
[6] Mahathir F. Sistem Pendeteksi Plagiat pada Dokumen Teks Berbahasa Indonesia Menggunakan Metode Rouge-N, Rouge-L, dan Rouge-W. Departemen Ilmu Komputer Institut Pertanian Bogor. Bogor, Indonesia. 2011.
[7] Soleman S, Purwarianti A. Tugas Akhir: Pengembangan Sistem Pendeteksi Plagiarisme pada Dokumen Berbahasa Indonesia. Bandung, Indonesia. 2012
[8] Alfikri Z, Purwarianti A. The Construction of Indonesian-English Cross Language Plagiarism Detection System Using Fingerprinting Technique. *Jurnal Ilmu Komputer dan Informasi, Vol 5, No 1*. Depok, Indonesia. 2012.
[9] Alzahrani S, Salim N, Abraham A. Understanding Plagiarism Linguistic Patterns, Textual Features and Detection Methods. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*. Taif, Arab Saudi. 2011.
[10] Potthast M, Stein B, Eiselt A, Barron-Cedeno A, Rosso P. Overview of the 1st International Competition on Plagiarism Detection. *SEPLN*. Donostia. 2009.
[11] Mitchell TM. Machine Learning. *The McGraw-Hill Companies, Inc.* 1997.
[12] Joachims T. *Text categorization with support vector machines: Learning with many relevant features*. Springer Berlin Heidelberg. 1998; 137-142.
[13] Sebastiani F. Machine learning in automated text categorization. *ACM computing surveys (CSUR)*. 2002; *34*(1): 1-47.
[14] Cortes C, Vapnik V. Support vector machine. *Machine learning*. 1995; *20*(3): 273-297.
[15] Rennie JD, Shih L, Teevan J, Karger D. *Tackling The Poor Assumptions of Naive Bayes Text Classifiers*. Machine Learning-International Workshop Then Conference. 2003; 20(2): 616.
[16] Stein B. Fuzzy-Fingerprints for Text-Based Information Retrieval. *Journal of Universal Computer Science.* 2005; 572-579.
[17] Schleimer S, Wilkerson D, Aiken A. Winnowing Local Algorithms for Document Fingerprinting. *SIGMOD 03 Proceedings of the 2003 ACM SIGMOD international conference on Management of data*. California. 2003.

[18] Salton G, Wong A, Yang CS. A Vector Space Model for Automatic Indexing. *Communications of the ACM.* 1975; 18(11): 613-620.
[19] Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*. 1990.