Two-Step Cluster based Feature Discretization of Naïve Bayes for Outlier Detection in Intrinsic Plagiarism Detection

Adi Wijaya
Graduate School of Informatics Engineering, STMIK Eresha
Email:adiwjj@gmail.com

Romi Satria Wahono
Faculty of Computer Science, Dian Nuswantoro University
Email: romi@brainmatics.com

Abstract: Intrinsic plagiarism detection is the task of analyzing a document with respect to undeclared changes in writing style which treated as outliers. Naive Bayes is often used to outlier detection. However, Naive Bayes has assumption that the values of continuous feature are normally distributed where this condition is strongly violated that caused low classification performance. Discretization of continuous feature can improve the performance of Naïve Bayes. In this study, feature discretization based on Two-Step Cluster for Naïve Bayes has been proposed. The proposed method using tf-idf and query language model as feature creator and False Positive/False Negative (FP/FN) threshold which aims to improve the accuracy and evaluated using PAN PC 2009 dataset. The result indicated that the proposed method with discrete feature outperform the result from continuous feature for all evaluation, such as recall, precision, f-measure and accuracy. The using of FP/FN threshold affects the result as well since it can decrease FP and FN; thus, increase all evaluation.

Keywords: intrinsic plagiarism detection, naïve bayes, feature discretization, two-step cluster

1 INTRODUCTION

The problem of plagiarism has recently increased because of the digital era of resources available on the web (Alzahrani, Salim, & Abraham, 2012). As a result, automated plagiarism analysis and detection receives increasing attention especially in academia (Maurer & Kappe, 2006). Intrinsic plagiarism detection (IPD), introduced by Meyer zu Eissen and Stein (2006), more ambitious since no reference corpus is given (Meyer zu Eissen, Stein, & Kulig, 2007) (Tschuggnall & Specht, 2012). IPD is a method for discovering plagiarism by analyzing a document with respect to undeclared changes in writing style (Stein, Lipka, & Prettenhofer, 2011). Since significant deviations in writing style are treated as outliers (Oberreuter & Velásquez, 2013); so, in IPD, outlier detection is important step.

Many studies have been published related to quantify writing style then detect its deviation writing style, such as using character n-gram profile as stylometric feature (Stamatatos, 2009b), word n-gram and word frequency (Oberreuter & Velásquez, 2013) and grammar tree as syntactical feature (Tschuggnall & Specht, 2012). Their approach still not produces excellent result due to unable to detect writing style change as outlier because writing style with small change (Stamatatos, 2009b), writing style change in short text (Oberreuter & Velásquez, 2013) and sentences with few words do not have a significant grammar tree and are therefore not detected (Tschuggnall & Specht, 2012).

This lack of outlier detection need to be solved and machine learning approach can be used for outlier detection (Chandola, Banerjee, & Kumar, 2009). One of algorithm to tackle this problem is Naïve Bayes (NB) since NB is often used to outlier, anomaly or novelty detection (Alan & Catal, 2011; Bahrepour, Zhang, Meratnia, & Havinga, 2009; Kamra, Terzi, & Bertino, 2007; Lepora et al., 2010). NB is fast, easy to implement with the simple structure, effective (Taheri & Mammadov, 2013). NB classifier continues to be a popular learning algorithm for data mining applications due to its simplicity and linear run-time (Hall, 2007).

However, NB has assumption that the values of continuous attributes are normally distributed within each class (Baron, 2014; Jamain & Hand, 2005; Soria, Garibaldi, Ambrogi, Biganzoli, & Ellis, 2011; Wong, 2012) where in many real-world data sets, this condition is strongly violated (Soria et al., 2011) and caused low classification performance (Yang & Webb, 2008).

With empirical evidence, discretization of continuous attributes can simplify data and improve the efficiency of inductive learning algorithms (Li, Deng, Feng, & Fan, 2011). Many discretization methods have been proposed to improve the performance of NB classifiers in terms of both time and accuracy (Ferreira & Figueiredo, 2012; Tsai, Lee, & Yang, 2008; Wong, 2012).

In this research, we propose the combination of Two-Step Cluster (TSC) and NB for improving the accuracy of outlier detection in IPD. TSC is applied to deal with the feature discretization of NB. TSC is chosen due to the ability to handle both continuous and categorical variables (Michailidou, Maheras, Arseni-Papadimititriou, Kolyva-Machera, & Anagnostopoulou, 2008; Satish & Bharadhwaj, 2010a), and in a single run, this procedure helps to identify the variables that significantly differentiate the segments from one another (Satish & Bharadhwaj, 2010a; Wu et al., 2006). TSC promises to solve at least some of these problems (e.g., the ability to deal with mixed-type variables and large data sets, automatic determination of the optimum number of clusters, and variables which may not be normally distributed) (Michailidou et al., 2008).

This paper is organized as follows. In section 2, the related works are explained. In section 3, the proposed method is presented. The experimental results of comparing the proposed method with others are presented in section 4. Finally, our work of this paper is summarized in the last section.

2 RELATED WORKS

Many studies have been published in which the IPD problem is further investigated both statistical based (Oberreuter & Velásquez, 2013; Stamatatos, 2009b; Tschuggnall & Specht, 2012) or machine learning based (Seaward & Matwin, 2009; Curran, 2010; Stein et al., 2011). To date, statistical based is dominated the research in IPD, but recently, machine learning based is trending since the result is promising.

Stamatatos (2009b) using character n-gram profile as stylometric feature that effective for quantifying writing style (Kanaris & Stamatatos, 2007; Koppel, Schler, & Argamon, 2009), robust to noisy text (Kanaris & Stamatatos, 2007) and language independent (Stamatatos, 2009a). This approach attempts to quantify the style variation within a document using character n-gram profiles and a style-change function based on an appropriate dissimilarity measure originally proposed for author identification. In the 1st International Competition on Plagiarism Detection 2009, this method was the first winner with precision, recall and overall are 0.2321, 0.4607 and 0.2462 respectively (Oberreuter & Velásquez, 2013).

Oberreuter & Velásquez (2013) showing that the usage of words can be analyzed and utilized to detect variations in style with great accuracy at the cost of detecting fewer cases (Oberreuter & Velásquez, 2013). They use word n-gram and word frequency as writing style quantification. In The 3rd International Competition on Plagiarism Detection 2011, they was the first winner with precision is 0.34, recall is 0.31 and overall is 0.33 (Oberreuter & Velásquez, 2013).

Tschuggnall & Specht (2012) using syntactical feature, namely the grammar used by an author, able to identify passages that might have been plagiarized due to the assumption is that the way of constructing sentences is significantly different for individual authors (Tschuggnall & Specht, 2012). They use grammar base as syntactical feature for writing style quantification and pq-gram-distance to identify the distance between two grammar trees. By comparing grammar trees and applying statistics the algorithm searches for significant different sentences and marks them as suspicious. The result is precision and recall value of about 32%.

Seaward & Matwin (2009) using three main components of its learning scheme, i.e. data pre-processors, learning algorithm and feature selector. In data pre-processor, the model use Kolmogorov Complexity (KC) measure as style feature. While in learning algorithm, it use Support Vector Machine (SVM) and NN and chi-square feature evaluator as feature selector. KC is used to describe the complexity or degree of randomness of a binary string and can be computed using any lossless compression algorithm. Their model use run-length encoding and Lempel-Ziv compression to create 10 complexity features, i.e. Adjective complexity, adjective count, global topic word complexity, verb word complexity, passive word complexity, active word complexity, preposition count, stop word count, average word length per sentence and local topic word complexity. Performance of the model shows that NN still better than SVM. NN outperform in precision (0.548) and f-measure (0.603) but lower recall (0.671) compared with SVM (recall=0.671, precision=0.521 and f-measure=0.587).

Curran (2010) using 3 components of its learning scheme. Data pre-processor is used in order to create an appropriate data feature that will feed to the model. Its main classifier is Neural Network (NN) and using Neuro-evolution of Augmenting Topologies (NEAT) as parameter optimizer of NN. The NEAT system evolves both NN structures and weights by incrementally increasing the complexity of NN. The model creates ten data features from their data pre-processor. A number of stylometric features were chosen for their study, such as: number of punctuation marks, sentence length (number of characters), sentence word frequency class, number of prepositions, number of syllables per word, average word length, number of stop-words, Gunning Fog index, Flesch index and Kincaid index. The model result 60% of accuracy for the plagiarized class, meaning that 60% of plagiarized sentences are recognized as being plagiarized.

Stein et al (2011) propose meta learning method as outlier post-processing and analyze the degradation in the quality of the model fitting process form its classifier, SVM. They use 3 kinds of stylometric features, such as lexical feature (character based), lexical feature (word based) and syntactic feature. In Meta learning method, they use 2 approaches, heuristic voting and unmasking. Heuristic voting is the estimation and use of acceptance and rejection thresholds based on the number of classified outlier sections and Unmasking measures the increase of a sequence of reconstruction errors, starting with a good reconstruction which then is successively impaired. The use of unmasking is considering a style outlier analysis as a heuristic to compile a potentially plagiarized and sufficiently large auxiliary document. The best result is using unmasking method as Meta learning. The result shows that collection of short plagiarized and light impurity has lowest precision (0.66), moderate recall (0.572) and moderate f-measure (0.67) among other collection while the best result is collection of long document and strong impurity with precision, recall and f-measure are 0.98, 0.60 and 0.74 respectively.

3 PROPOSED METHOD

We propose a method called TSC-FD+NB, which is short for Two-Step Cluster based feature discretization for Naïve Bayes (NB) to achieve better detection performance of outlier detection in intrinsic plagiarism detection. The proposed method evaluated using dataset PAN PC 2009 and using term weighting (tf-idf) and 3 functions in query language model (QLM) based on Ponte and Croft (1998) model, i.e.: mean term frequency of term in documents where it is occurs, risk for term in document and probability of producing the query for given document as document quantification. Figure 1 shows block diagram of the proposed method.

The aim of discretization by using Two-Step Cluster (TSC) is to improve NB classification performance since its tends to be better when continuous features are discretized (Dougherty, 1995). Discretization is a popular approach to handling continuous feature in machine learning (Yang & Webb, 2002); discretization of continuous features can simplify data, more compact and improve the efficiency of inductive learning algorithms. TSC is one of clustering algorithm that developed firstly by Chiu et al. (2001) and designed to handle very large data sets, is provided by the statistical package SPSS. Two-Step cluster able to handle both continuous and categorical variables (Michailidou et al., 2008; Satish & Bharadhwaj, 2010a).

As shown in Figure 1, after processed dataset which is dataset have been quantified is feeding, TSC is used for create class label from processed data. This class label is consisting of two labels, 1 and 2 which mean 1 for plagiarized term and 2 for plagiarism-free term; since there is no information about class label for each term. After class label is defined, prior probability of each class label for NB calculation is conducted which is needed in NB calculation. The next step is feature discretization where each features discrete with 4 classes. The idea is the same with create 4-binning or 4 quartile in equal width or equal frequency interval binning as unsupervised discretization method. After that, NB calculation for discrete feature is conducted and resulting status of each term with four statuses may be resulted, i.e.: true positive (TP), true negative (TN), false positive (FP) and false negative (FN). If the status is FP or FN, so the next activity is status adjustment following condition as shown in Table 1. If status is not FP or not FN, final status is the same with previous status.

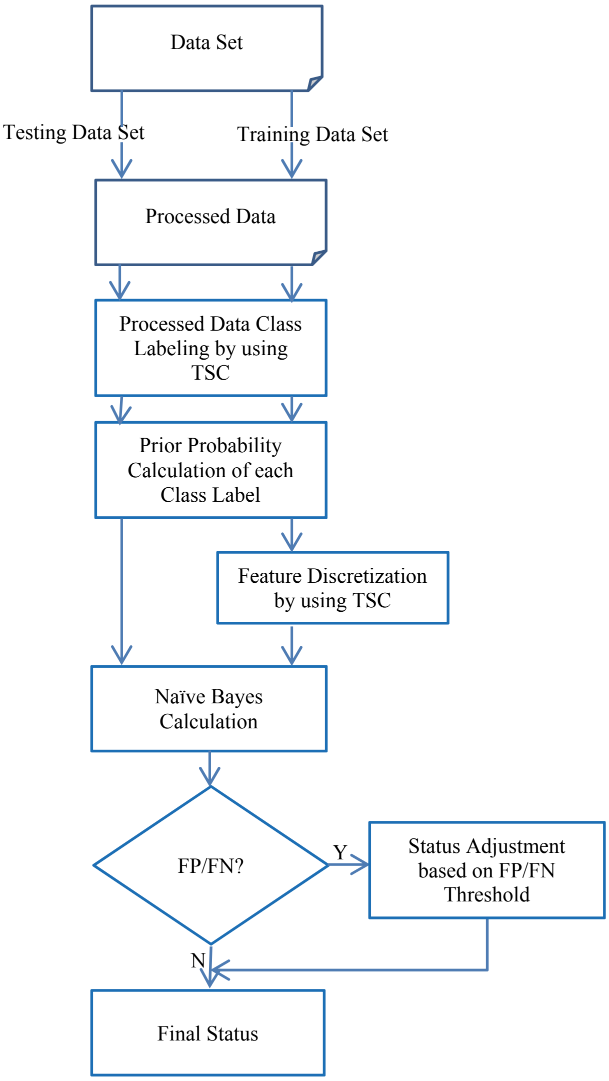
Data Set
Testing Data Set
Training Data Set
Processed Data
Processed Data Class Labeling by using TSC
Prior Probability Calculation of each Class Label
Feature Discretization by using TSC
Naïve Bayes Calculation
FP/FN?
Y
Status Adjustment based on FP/FN Threshold
N
Final Status

Figure 1. Block Diagram of TSC-FD+NB Method

Table 1. Status Adjustment Condition

First Status	Threshold Comparison	Adjusted Guess	Adjusted Status
FP	$[P(C_2\|x) > P(C_1\|x)] <=$ Threshold	1	TN
	$[P(C_2\|x) > P(C_1\|x)] >$ Threshold	2	FP
FN	$[P(C_1\|x) > P(C_2\|x)] <=$ Threshold	2	TP
	$[P(C_1\|x) > P(C_2\|x)] >$ Threshold	1	FN

In this study, the evaluation method using the classifier's effectiveness (Oberreuter & Velásquez, 2013). The results of this process will produce a confusion matrix that contains the value true positive (TP), true negative (TN), false positive (FP) and false negative (FN) as shown in Table 2. The main result is model accuracy and common information retrieval measurement, such as: recall, precision and f-measure. It calculated based on confusion matrix that produces from the model. Based on confusion matrix, the measurement calculation are as follows:

$$Recall = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$F - Measure = \frac{2\ x\ Precision\ x\ Recall}{Precison + Recall}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Table 2. Confusion Matrix

	Actual	
Guess	Plagiarized	Plagiarism-free
Plagiarized	TP	FP
Plagiarism-free	FN	TN

The main result which is accuracy of all the models will be compared with statistical test both parametric and non-parametric test. T-test as one of parametric test is used to compare between two models if their sample distribution is normal or Wilcoxon Signed Ranks test, one of non-parametric test, if their sample distribution is not normal. While comparison between multi models using Friedman test, one of non-parametric test, to verify whether there is a significant difference between the proposed methods as Demsar (2006) suggested Friedman test for multi classifier or model comparisons. After that, post-hoc test conducted using Nemenyi Test to detect which models significantly has different result since Friedman Test only show whether there is different or no. Therefore, parametric test and non-parametric test are used in this study.

4 EXPERIMENTAL RESULTS

The experiments are conducted using a computing platform based on Intel Core 2 Duo 2.2 GHz CPU, 2 GB RAM, and Microsoft Windows XP SP2 32-bit operating system. The development environment is MS Visual Basic 6, PHP and MySql as database server.

First of all, we conducted experiments on PAN PC 2009 with continuous feature. The experimental results are reported in Table 3. NB with continuous feature perform not so good since has low in all result. FN/FN threshold doesn't change the recall since the difference between posterior still higher than FP/FN thresholds.

Table 3. Results on NB with Continuous Feature

FP/FN Threshold	R	P	F	Accuracy
0	0.167	0.061	0.090	0.788
0.001	0.167	0.062	0.090	0.790
0.005	0.167	0.063	0.091	0.793
0.01	0.167	0.064	0.092	0.794
0.05	**0.167**	**0.071**	**0.099**	**0.811**

In the next experiment, we implemented NB with discrete feature on PAN PC 2009 dataset. The experimental result is shown in Table 4. The improved model is highlighted with boldfaced print. NB with discrete feature model perform excellent rather than continuous feature. FP/FN threshold also increase both recall and precision; thus, F-measure and accuracy increase as well.

The result of comparison of recall, precision, f-measure and accuracy can be described in Figure 2, Figure 3, Figure 4 and Figure 5 respectively. As shown in Figure 2, recall of NB with discrete feature overcome recall in continuous feature. Recall in NB with discrete feature linearly increasing when FP/FN threshold increase. Although NB with discrete feature recall in FP/FN Threshold = 0 is lower than continuous feature, as the increase in the FP/FN threshold, NB with discrete feature recall is increase and overcome the continuous feature with the best recall is 0.247.

Table 4. Results on NB with Discrete Feature

FP/FN Threshold	R	P	F	Accuracy
0	0.156	0.057	0.083	0.786
0.001	0.172	0.073	0.102	0.812
0.005	0.182	0.119	0.144	0.865
0.01	0.186	0.187	0.186	0.899
0.05	**0.247**	**1.000**	**0.397**	**0.953**

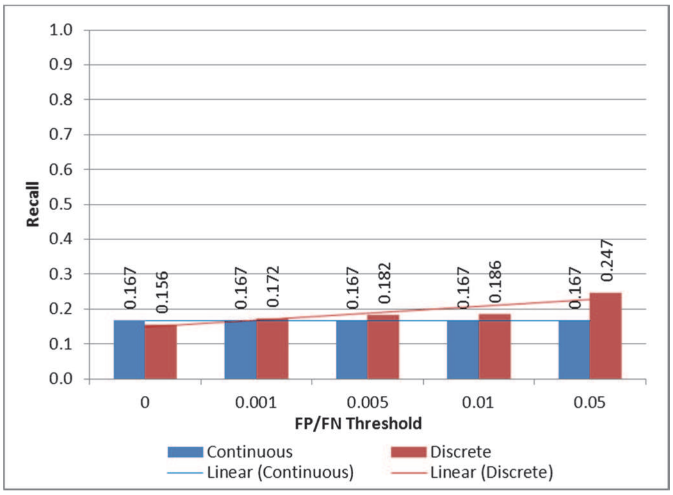

Figure 2. NB with Continuous vs. Discrete Feature's Recall

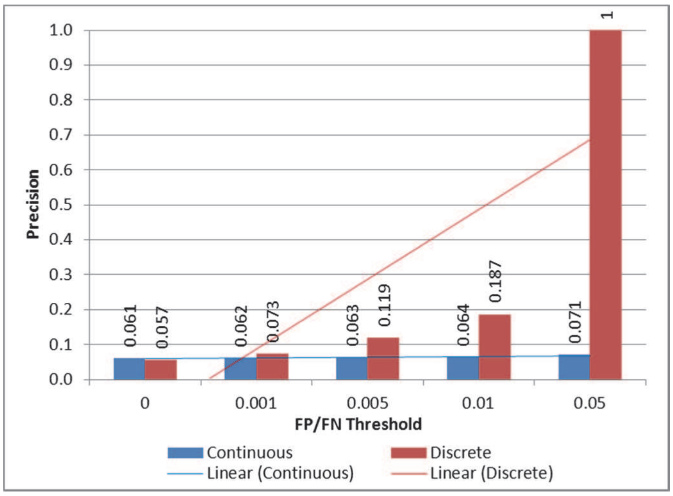

Figure 3. NB with Continuous vs. Discrete Feature's Precision

As shown in Figure 3, precision of NB with discrete feature is overcome continuous feature in almost models except model with FP/FN Threshold = 0. Precision in NB with discrete feature linearly increase when FP/FN threshold increase with the best precision is 1 because in this model value of FP=0. As shown in Figure 3, precision of NB with discrete feature increased sharply compared to continuous feature.

F-measure is harmonic mean between recall and precision. As shown in Figure 4, f-measure of NB with continuous feature tends to constant while f-measure of NB with discrete feature is increases as FP/FN threshold increase. This is because recall and precision of NB with discrete feature increase sharply compared to continuous feature. The best result is model with FP/FN Threshold = 0.05 both NB with continuous feature and NB with discrete feature. The rising of F-measure of NB with continuous feature about 0.109 times from the lowest result model with FP/FN threshold while in NB with discrete feature, it rises 3.754 times from the lowest result. This because precision of NB with discrete feature increase sharply compared to continuous feature.

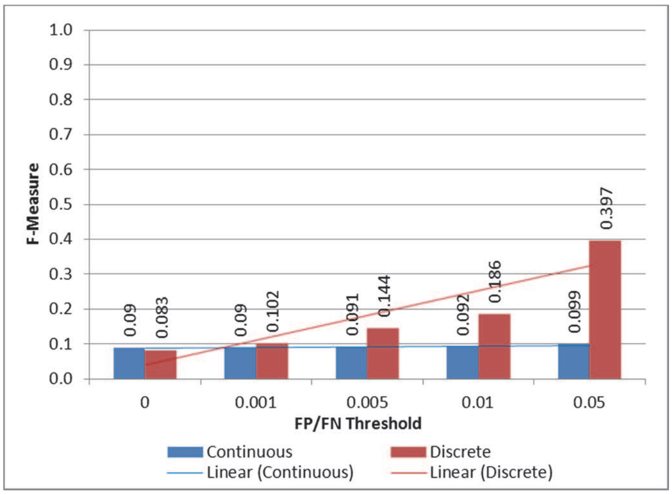

Figure 4. NB with Continuous vs. Discrete Feature‘s F-measure

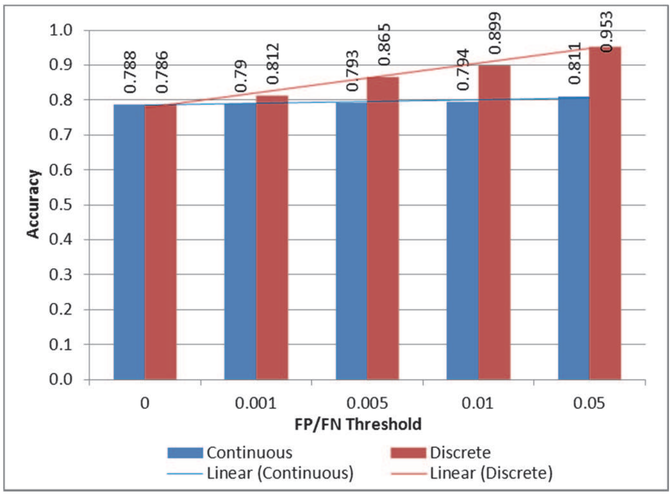

Figure 5. NB with Continuous vs. Discrete Feature‘s Accuracy

As shown in Figure 5, accuracy of NB with discrete feature is overcome NB with continuous feature with linear increment while accuracy of NB with continuous feature tent to constant. The best accuracy is 0.953 reached by model of NB with discrete feature with FP/FN threshold 0.05. Accuracy of NB with continuous feature increases only 0.029 times from the lowest result, model with FP/FN Threshold = 0 while in NB with discrete feature, it increases about 0.212 times.

This result indicates that FP/FP threshold is effective enough to increase all measurement in NB with discrete feature while not effective in NB with continuous feature. Threshold is one of scheme that many researchers used to improve plagiarism detection such as in case the similarity score, if above a threshold, the detected plagiarism case is considered true and otherwise (Stamatatos, 2011b); lowering FP value so less time is required to filter out FPs by the evaluator (Chen et al., 2010) and for the last purpose is obtaining a reasonable trade-off between precision and recall (Oberreuter & Velásquez, 2013). In this study, the result also confirm some studies that recall is tend to lower for short document since false negatives are relatively short documents (Stamatatos, 2009b) and also recall still low for document with little portion of plagiarizes passage (Oberreuter & Velásquez, 2013).

The results of both methods are compared in order to verify whether a significant difference between NB with continuous feature and the proposed TSC-FD+NB method. We compare between methods with the same FP/FN thresholds. Since there is model with no normal distribution of data, so comparison is used Wilcoxon signed ranks test. Table 5 shows the result.

Table 5. Wilcoxon Signed Rank Test for Continuous Feature’s Accuracy vs. Discrete Feature’s Accuracy

	Negative Ranks	Positive Ranks	Z	Asymp. Sig. (2-tailed)
c_th_0 vs d_th_0	13.53 (20/30)	18.28 (9/30)	-1.147	**0.252**
c_th_0.001 vs. d_th_0.001	15.66 (25/30)	10.88 (4/30)	-3.763	**0**
c_th_0.005 vs. d_th_0.005	15 (29/30)	0 (0/30)	-4.703	**0**
c_th_0.01 vs. d_th_0.01	15.00 (29/30)	0 (0/30)	-4.703	**0**
c_th_0.05 vs. d_th_0.05	15 (29/30)	0 (0/30)	-4.703	**0**

As shown in Table 5, there is 5 pair of comparisons. In Pair 1, NB with continuous feature with FP/FN Threshold = 0 (c_th_0) is compare to NB with discrete feature with FN/FP Threshold = 0.001 (d_th_0). P-value = 0.252 (> 0.005), so H_0 is failed to be rejected. This indicates that there is no difference accuracy between c_th_0 and d_th_0. Another consideration is negative ranks and positive ranks, which indicates that d_th_0 accuracy is higher than c_th_0 accuracy with 20 cases out of 30 cases (negative ranks) while positive ranks is 9 cases of 30 cases, which means there is 9 cases in c_th_0 that its accuracy is higher than d_th_0.

In Pair 2, NB with continuous feature with FP/FN Threshold = 0.001 (c_th_0.001) is compare to NB with discrete feature with FP/FN Threshold = 0.001 (d_th_0.001). P-value = 0.000 (<0.005), so H_0 is rejected. This indicates that there is difference accuracy between c_th_0.001 and d_th_0.001. Another consideration is negative ranks and positive ranks, which indicates that d_th_0.001 accuracy is higher than c_th_0.001 accuracy with 25 cases out of 30 cases (negative ranks) while positive ranks is 4 cases of 30 cases, which means there is only 4 cases in c_th_0.001 that its accuracy is higher than d_th_0.001. Another pairs are having the same result. All of them indicate that there is difference accuracy where p-value = 0.000 and negative ranks is 29 cases out of 30 cases, which means accuracy of 29 cases in d_th_0.005, d_th_0.01 and d_th_0.05 are higher than c_th_0.005, c_th_0.01 and c_th_0.05 respectively.

Finally, Demsar recommends the Friedman test for classifier comparisons, which relies on less restrictive assumptions (Demsar, 2006). Based on this recommendation, the Friedman test is employed in this study to compare the accuracy of the different models. At first comparison, NB with various FP/FN thresholds are tested with Friedman test both continuous feature and discrete feature. Table 6 shows Friedman test result.

As shown in Table 6, p-value < 0.0001 which is smaller than significance level (α = 0.05). The null hypothesis, H_0 is that the samples come from the same population. Since p-value < 0.05, so H_0 is rejected. This means that sample comes from different population. It indicates that there is difference between models, but Friedman test doesn’t provide which model is different. To answer that question, post-hoc test is used, which is in this case is Nemenyi test. According to the Nemenyi test the performance of two models is significantly different if the corresponding mean ranks differ by at least the critical difference (Li et al., 2011). Nemenyi test is similar to the Tukey test for ANOVA and is used when all classifiers are compared to each other (Demsar, 2006).

Table 6. Friedman Test of All Models Accuracy

Q (Observed value)	234.3890
Q (Critical value)	16.9190
DF	9
p-value (Two-tailed)	< 0.0001
alpha	0.05

Table 7. Multiple pairwise comparisons using Nemenyi's procedure of All Models Accuraccy

Sample*)	Frequency	Sum of ranks	Mean of ranks
(1) c_0	30	43.5000	1.4500
(2) c_0.001	30	81.5000	2.7167
(3) d_0	30	106.5000	3.5500
(4) c_0.005	30	111.5000	3.7167
(5) c_0.01	30	134.0000	4.4667
(6) d_0.001	30	178.0000	5.9333
(7) c_0.05	30	194.0000	6.4667
(8) d_0.005	30	237.0000	7.9000
(9) d_0.01	30	267.5000	8.9167
(10) d_0.05	30	296.5000	9.8833

Table 7 described the mean rank based procedures Nemenyi. The mean of ranks is obtained from the comparison between the models, the higher the rank, the higher the point, and then divided by the number of data samples. The Nemenyi test calculates all pairwise comparisons between different models and checks which models' performance differences exceed the critical difference (CD), which are 2.4732.

The significant difference table of the Nemenyi test is shown in Table 8. Model of NB with discrete feature with FP/FN threshold is the most different with other models about 7 differences. In continuous feature, model of NB with FP/FN threshold 0.005 and 0.01 have fewest difference with other models with 3 differences. In discrete feature, model of NB with FP/FN threshold 0 and 0.001 have fewest difference with other models with 3 differences as well.

Table 8. Significant Differences of All Models Accuracy

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
(1)	N	N	N	**Y**	**Y**	N	**Y**	**Y**	**Y**	**Y**
(2)	N	N	N	N	**Y**	N	**Y**	**Y**	**Y**	**Y**
(3)	N	N	N	N	**Y**	N	N	**Y**	**Y**	**Y**
(4)	**Y**	N	N	N	N	N	N	**Y**	**Y**	**Y**
(5)	**Y**	**Y**	**Y**	N	N	**Y**	N	N	N	**Y**
(6)	N	N	N	N	**Y**	N	N	**Y**	**Y**	**Y**
(7)	**Y**	**Y**	N	N	N	N	N	N	**Y**	**Y**
(8)	**Y**	**Y**	**Y**	**Y**	N	**Y**	N	N	N	N
(9)	**Y**	**Y**	**Y**	**Y**	N	**Y**	**Y**	N	N	N
(10)	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	**Y**	N	N	N

As shown in Table 7 and 8, model of NB with discrete feature with FP/FN Threshold = 0.05 outperform other models followed by model of NB with discrete feature with FP/FN Threshold = 0.01, model of NB with discrete feature with FP/FN Threshold = 0.005, model of NB with continuous feature with FP/FN Threshold = 0.05 and model of NB with discrete feature with FP/FN Threshold = 0.001 in the second, third, fourth and fifth rank respectively. Overall model of NB with discrete feature outperform model of NB with continuous feature. This result confirmed Webb (2001), that explain why discretization can be effective for NB classifier.

These results prove that NB with discrete feature is better than NB with continuous feature. The discretization, referring to Yang and Webb (2003) can be effective for NB classifier. TSC as the method of discretization is promising method because support both continuous and categorical variables, and in a single run; this procedure helps to identify the variables that significantly differentiate the segments from one another (Satish & Bharadhwaj, 2010b), automatic determination of the optimum number of clusters, and variables which may not be normally distributed (Michailidou et al., 2008). From this study, TSC can be alternative as discretization based on the clustering analysis, such as k-means discretization (Dash, Paramguru, & Dash, 2011; Richhariya & Sharma, 2014) and shared nearest neighbor clustering algorithm (Gupta, Mehrotra, & Mohan, 2010). Thus, TSC based discretization of NB can be used to improve the performance of intrinsic plagiarism detection by detect outlier from short plagiarized passage in a document.

5 CONCLUSION

The experimental result shows that the result models of NB with discrete feature outperform the result from NB with continuous feature for all measurement, such as recall, precision, f-measure and accuracy with significant difference. The using of FP/FN threshold affect the result as well with FP/FN threshold = 0.05 is the best since it can decrease false positive (FP) and false negative (FN) better than other value in all models especially model with discrete feature where the decrement of FP and FN is highest; thus, increase all measurement especially precision and accuracy. Therefore, it can be concluded that feature discretization based on Two-Step Cluster can improve the accuracy of NB for outlier detection in intrinsic plagiarism detection compared to NB with continuous feature.

REFERENCES

Alan, O., & Catal, C. (2011). Thresholds based outlier detection approach for mining class outliers: An empirical case study on software measurement datasets. *Expert Systems with Applications*, *38*(4), 3440–3445.

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

Bahrepour, M., Zhang, Y., Meratnia, N., & Havinga, P. J. M. (2009). Use of event detection approaches for outlier detection in wireless sensor networks. *2009 International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, 439–444.

Baron, G. (2014). Influence of Data Discretization on Efficiency of Bayesian Classifier for Authorship Attribution. *Procedia Computer Science*, *35*, 1112–1121.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, *41*(3), 1–72.

Chen, C., Yeh, J., & Ke, H. (2010). Plagiarism Detection using ROUGE and WordNet, *2*(3), 34–44.

Chiu, T., Fang, D., Chen, J., Wang, Y., & Jeris, C. (2001). A Robust and Scalable Clustering Algorithm for Mixed Type Attributes in Large Database Environment. In *Proceedings of the 7th ACM SIGKDD Internation- al Conference on Knowledge Discovery and Data Mining* (pp. 263–268).

Curran, D. (2010). An evolutionary neural network approach to intrinsic plagiarism detection. In *AICS 2009, LNAI 6206* (pp. 33–40). Springer-Verlag Berlin Heidelberg.

Dash, R., Paramguru, R. L., & Dash, R. (2011). Comparative Analysis of Supervised and Unsupervised Discretization Techniques, *2*(3), 29–37.

Demsar, J. (2006). Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, *7*, 1–30.

Dougherty, J. (1995). Supervised and Unsupervised Discretization of Continuous Features.

Ferreira, A. J., & Figueiredo, M. a. T. (2012). An unsupervised approach to feature discretization and selection. *Pattern Recognition*, *45*(9), 3048–3060.

Gupta, A., Mehrotra, K. G., & Mohan, C. (2010). A clustering-based discretization for supervised learning. *Statistics & Probability Letters*, *80*(9-10), 816–824. doi:10.1016/j.spl.2010.01.015

Hall, M. (2007). A decision tree-based attribute weighting filter for naive Bayes. *Knowledge-Based Systems*, *20*(2), 120–126.

Jamain, A., & Hand, D. J. (2005). The Naive Bayes Mystery: A classification detective story. *Pattern Recognition Letters*, *26*(11), 1752–1760.

Kamra, A., Terzi, E., & Bertino, E. (2007). Information Assurance and Security Detecting anomalous access patterns in relational databases.

Kanaris, I., & Stamatatos, E. (2007). Webpage Genre Identification Using Variable-Length Character n-Grams. In *19th IEEE International Conference on Tools with Artificial Intelligence(ICTAI 2007)* (pp. 3–10). IEEE.

Koppel, M., Schler, J., & Argamon, S. (2009). Computational methods in authorship attribution. *Journal of the American Society for Information Science and Technology*, *60*(1), 9–26.

Lepora, N. F., Pearson, M. J., Mitchinson, B., Evans, M., Fox, C., Pipe, A., Prescott, T. J. (2010). Naive Bayes novelty detection for a moving robot with whiskers. *2010 IEEE International Conference on Robotics and Biomimetics*, 131–136.

Li, M., Deng, S., Feng, S., & Fan, J. (2011). An effective discretization based on Class-Attribute Coherence Maximization. *Pattern Recognition Letters*, *32*(15), 1962–1973.

Maurer, H., & Kappe, F. (2006). Plagiarism - A Survey, *12*(8), 1050–1084.

Meyer zu Eissen, S., Stein, B., & Kulig, M. (2007). Plagiarism Detection Without Reference Collections. In *Studies in Classification, Data Analysis, and Knowledge Organization, Advances in Data Analysis* (pp. 359–366). Berlin: Springer.

Michailidou, C., Maheras, P., Arseni-Papadimititriou, a., Kolyva-Machera, F., & Anagnostopoulou, C. (2008). A study of weather types at Athens and Thessaloniki and their relationship to circulation types for the cold-wet period, part I: two-step cluster analysis. *Theoretical and Applied Climatology*, *97*(1-2), 163–177.

Oberreuter, G., & Velásquez, J. D. (2013). Text mining applied to plagiarism detection: The use of words for detecting deviations in the writing style. *Expert Systems with Applications*, *40*(9), 3756–3763.

Ponte, J. M., & Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 275–281).

Richhariya, V., & Sharma, N. (2014). Optimized Intrusion Detection by CACC Discretization Via Naïve Bayes and K-Means Clustering, *14*(1), 54–58.

Satish, S. M., & Bharadhwaj, S. (2010a). Information Search Behaviour among New Car Buyers: A Two-Step Cluster Analysis. *IIMB Management Review*, *22*(1-2), 2.

Satish, S. M., & Bharadhwaj, S. (2010b). Information search behaviour among new car buyers: A two-step cluster analysis. *IIMB Management Review*, *22*(1-2), 5–15.

Seaward, L., & Matwin, S. (2009). Intrinsic Plagiarism Detection using Complexity Analysis. In *SEPLN 2009 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN 2009)* (pp. 56–61).

Soria, D., Garibaldi, J. M., Ambrogi, F., Biganzoli, E. M., & Ellis, I. O. (2011). A "non-parametric" version of the naive Bayes classifier. *Knowledge-Based Systems*, *24*(6), 775–784.

Stamatatos, E. (2009a). A survey of modern authorship attribution methods. *Journal of the American Society for Information Science and Technology*, *60*(3), 538–556.

Stamatatos, E. (2009b). Intrinsic Plagiarism Detection Using Character n -gram Profiles. In *Stein, B., Rosso, P., Stamatatos, E., Koppel, M., Agirre, E. (eds.) SEPLN 2009 Workshop on Uncovering Plagiarism, Authorship, and Social Software Misuse (PAN 2009)* (pp. 38–46).

Stamatatos, E. (2011). Plagiarism Detection Using Stopword n -grams, *62*(12), 2512–2527.

Stein, B., Lipka, N., & Prettenhofer, P. (2011). Intrinsic plagiarism analysis. *Language Resources and Evaluation*, *45*(1), 63–82.

Taheri, S., & Mammadov, M. (2013). Learning the naive Bayes classifier with optimization models. *International Journal of Applied Mathematics and Computer Science*, *23*(4), 787–795.

Tsai, C.-J., Lee, C.-I., & Yang, W.-P. (2008). A discretization algorithm based on Class-Attribute Contingency Coefficient. *Information Sciences*, *178*(3), 714–731.

Tschuggnall, M., & Specht, G¨. (2012). Plag-Inn: Intrinsic Plagiarism Detection Using Grammar Trees. In G. Bouma, A. Ittoo, E. Métais, & H. Wortmann (Eds.), *LNCS-Natural Language Processing and Information Systems* (Vol. 7337, pp. 284–289). Berlin, Heidelberg: Springer Berlin Heidelberg.

Webb, G. I. (2001). On Why Discretization Works for Naive-Bayes Classifiers.

Wong, T.-T. (2012). A hybrid discretization method for naïve Bayesian classifiers. *Pattern Recognition*, *45*(6), 2321–2325.

Wu, H., Zhang, X., Li, X., Liao, P., Li, W., Li, Z., … Pei, F. (2006). Studies on Acute Toxicity of Model Toxins by Proton Magnetic Resonance Spectroscopy of Urine Combined with Two-step Cluster Analysis. *Chinese Journal of Analytical Chemistry*, *34*(1), 21–25.

Yang, Y., & Webb, G. I. (2002). A Comparative Study of Discretization Methods for Naive-Bayes Classifiers. In *Proceedings of PKAW 2002, The 2002 Pacific Rim Knowledge Acquisition Work- shop* (pp. 159–173). Tokyo, Japan.

Yang, Y., & Webb, G. I. (2008). Discretization for naive-Bayes learning: managing discretization bias and variance. *Machine Learning*, *74*(1), 39–74.

BIOGRAPHY OF AUTHORS

Adi Wijaya. Received MKom from STMIK Eresha, Jakarta. He is an IT professional and part-time lecturer at STIKIM, Jakarta. His current research interests include information retrieval and machine learning.

Romi Satria Wahono. Received B.Eng and M.Eng degrees in Computer Science respectively from Saitama University, Japan, and Ph.D in Software Engineering from Universiti Teknikal Malaysia Melaka. He is a lecturer at the Graduate School of Computer Science, Dian Nuswantoro University, Indonesia. He is also a founder and chief executive officer of Brainmatics, Inc., a software development company in Indonesia. His current research interests include software engineering and machine learning. Professional member of the ACM, PMI and IEEE Computer Society.