Feature Selection Methods for Writer Identification: A Comparative Study

Satrya Fajri Pratama
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Melaka, Malaysia
rascove@yahoo.com

Azah Kamilah Muda
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Melaka, Malaysia
azah@utem.edu.my

Yun-Huoy Choo
Faculty of Information and Communication Technology
Universiti Teknikal Malaysia Melaka
Melaka, Malaysia
huoy@utem.edu.my

***Abstract*—Feature selection is an important area in the machine learning, specifically in pattern recognition. However, it has not received so many focuses in Writer Identification domain. Therefore, this paper is meant for exploring the usage of feature selection in this domain. Various filter and wrapper feature selection methods are selected and their performances are analyzed using image dataset from IAM Handwriting Database. It is also analyzed the number of features selected and the accuracy of these methods, and then evaluated and compared each method on the basis of these measurements. The evaluation identifies the most interesting method to be further explored and adapted in the future works to fully compatible with Writer Identification domain.**

Keywords-feature selection; filter method; wrapper method; writer identification; comparative study

I. INTRODUCTION

Feature selection has become the focus of research area for a long time. The purpose of feature selection is to obtain the most minimal sized subset of features as long as the classification accuracy does not significantly decreased and the result of the selected features class distribution is as close as possible to original class distribution [1].

In theory, more features produce more discriminating power. However, practical experience has shown that this is not always the case. If there is too much irrelevant and redundant information present, the performance of a classifier might be degraded. Removing these irrelevant and redundant features can improve the classification accuracy.

The three popular methods of feature selection are filter method, wrapper method, and embedded method, as shown in Fig. 1 that has been presented in [2]. Filter method assesses the relevance of features by looking only at the intrinsic properties of the data. A feature relevance score is calculated, and low-scoring features are removed [3]. Simultaneously, wrapper method uses an induction algorithm to estimate the merit of feature subsets. It explores the space of features subsets to optimize the induction algorithm that uses the subset for classification [4]. On the other hand, in embedded method, the selection process is done inside the induction algorithm itself, being far less computationally intensive compared with wrapper methods [5].

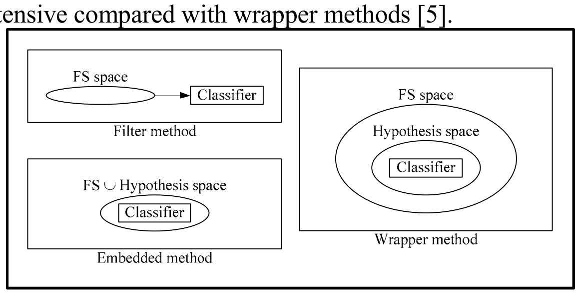


Figure 1. Feature selection models

Studies have shown that there are no feature selection methods more superior compared to others [6]. Which methods to use sometimes depends on the size of the data itself. Using filter methods means to have a good computational complexity, but the higher complexity of the wrapper methods will also produce higher accuracy in the final result, whereas embedded methods are intrinsic to some learning algorithm and so only those algorithm designed with this characteristic can be used.

Writer Identification (WI) is an active area of research in pattern recognition due to extensive exchange of paper documents, although currently the world has already moved toward the use of digital documents. WI distinguishes writers based on the handwriting, and ignoring the meaning of the words. Previous studies have explored various methods to improve WI domain, and these studies produced the satisfying performance. However, the use of feature selection as one of important machine learning task is often disregarded in WI domain, with only a handful of studies implemented feature selection task in the WI domain, for instance the studies conducted by [7], [8], and [9].

The goal of this paper is to evaluate the performance of various filter and wrapper feature selection methods on some small-sized data sets, where the number of features is between 1-19 features [10], specifically for WI domain. This is also motivated by facts revealed by the scarce focus in exploring the usage of feature selection in WI domain.

978-1-4244-8950-3/10/$26.00 ©2010 IEEE

The remainder of this paper is organized into five sections. In Section II, WI domain is briefly explained. Section III explains each feature selection algorithms chosen and used. In Section IV and V, experimental setup is presented and the results are shown. In the last section, the study is concluded and the future works are proposed.

II. WRITER IDENTIFICATION

The individuality of writer lies in the hypothesis that everyone has consistent handwriting that is different from another individual [11]. The individuality of handwriting enables the identification process of the writer of particular handwriting through scientific analysis.

Handwriting analysis consists of two categories, which are handwriting recognition and handwriting identification. Handwriting recognition deals with the contents conveyed by the handwritten word, while handwriting identification tries to differentiate handwritings to determine the author. There are two tasks in identifying the writer of handwriting, namely identification and verification. Identification task determines the writer of handwriting from many known writers, while verification task determines whether one document and another is written by the same writer.

WI has attracted many researchers to work in, primarily in forensic and biometric applications. The challenge is to find the best solution to identify the writer accurately; therefore the main issue is how to acquire the individual features from various handwritings, and thus various studies have been conducted and discussed in [12].

Bensefia, Nosary, Paquet and Heutte in [13] mentioned that in traditional handwriting identification task, there are three steps involved, which are pre-processing, feature extraction and classification. Previous studies have explored various methods to enhance traditional task, and improves the classification accuracy. One study has been conducted by Muda [13] by incorporating discretization task after feature extraction task, and the results shows significantly improved classification accuracy.

This paper extends the work of Muda [13] with a slight modification. This paper tries to incorporate feature selection after feature extraction task, instead of using discretization task. It is expected after the feature selection task, the classification result will improve, or similar to original set.

III. FEATURE SELECTION METHODS

In this paper, six feature selection methods retrieved from various literatures are chosen. Five of them are filter method, while the last one is wrapper method, which uses five different classifiers. All methods are available in public domain.

A. Filter Methods

Five filter methods feature selection are considered, which are ReliefF, Correlation-based Feature Selection (CFS), Consistency-based Feature Selection, also known as Las Vegas Algorithm Filter Version (LVF), Fast Correlation-based Filter (FCBF), and Significance Measurement Feature Selection (SMFS).

1) ReliefF

ReliefF is an extension of Relief that is first introduced by Kira and Rendell [14]. The basic idea is to measure the relevance of features in the neighborhoods around target samples. ReliefF finds the nearest sample in feature space of the same category, called the "hit" sample, then measures the distance between the target and hit samples. It also finds the nearest sample of the other category, called the "miss" sample, and then does the same work. ReliefF uses the difference between those measured distances as the weight of target feature. The contribution of all the hits and all the misses are then averaged. The contribution for each class of the misses is weighted with the prior probability of that class *P(C)*.

2) CFS

CFS ranks feature subsets according to a correlation based heuristic evaluation function [2]. CFS selects subsets that contain highly correlated features with the class and uncorrelated with each other. The acceptance of a feature will depend on the extent to which it predicts classes in areas of the instance space not already predicted by other features. The feature subset evaluation function is

$$M_s = \frac{k\overline{r_{cf}}}{\sqrt{k+k(k-1)\overline{r_{ff}}}} \tag{1}$$

where M_S is the heuristic "merit" of a feature subset S containing k features, $\overline{r_{cf}}$ is the mean feature-class correlation ($f \in S$), and $\overline{r_{ff}}$ is the average feature-feature inter-correlation.

CFS calculates the correlations and then searches the feature subset space. The subset with the highest merit found is used to reduce the dimensionality.

3) LVF

LVF uses a random generation of subsets and an inconsistency measure as evaluation function [15]. Two instances are inconsistent if they have the same feature values but different classes. The inconsistency measure of a given subset of features T relative to a dataset D is defined as

$$Inconsistency(T, D) = \frac{\sum_{i=1}^{K}|D_i| - |M_i|}{N} \tag{2}$$

where $|D_i|$ is the number of occurrences of the i-th feature value combination on T, K is the number of the distinct combinations of features values on T, $|M_i|$ is the cardinality of the class to which belong the majority of instances on the i-th feature value combination, and N is the number of instances in the dataset D.

The algorithm requires an inconsistency threshold close or equal to zero. Any candidate subset having is rejected if inconsistency greater than the threshold. When the maximum number of generated subsets is reached, the generation process is stopped.

4) FCBF

FCBF uses Symmetrical Uncertainty (SU) to calculate dependences of features and finds best subset using backward selection technique with sequential search strategy [16]. SU is a normalized information theoretic measure which uses entropy and conditional entropy values to calculate dependencies of features. If X is a random variable and $P(x)$ is the probability of x, the entropy of X is

$$H(X) = -\sum_i P(x_i) log_2(P(x_i)) \quad (3)$$

Conditional entropy or conditional uncertainty of X given another random variable Y is the average conditional entropy of X over Y

$$H(X|Y) = -\sum_j P(y_j) \sum_i P(x_i|y_j) log_2(P(x_i|y_j)) \quad (4)$$

$$SU(X,Y) = 2\left[\frac{IG(X|Y)}{H(X)+H(Y)}\right] \quad (5)$$

An SU value of 1 indicates that using one feature other feature's value can be totally predicted and value 0 indicates two features are totally independent. The SU values are symmetric for both features.

5) SMFS

SMFS computes the significance measure based on the rationale that a significant feature is likely to have different values for different classes [17]. The relative frequency of an attribute value across different classes gives a measure of the attribute value-to-class and class-to-attribute value associations. These associations are stored and form a part of classificatory knowledge.

For each attribute A_i, this algorithm computes the overall attribute-to-class association denoted by Æ(A_i) with k different attribute values. Æ(A_i) lies between 0.0 and 1.0, and is computed as

$$Æ(A_i) = \left(\frac{1}{k}\sum_{r=1}^{k} \vartheta_i^r\right) - 1.0 \quad (6)$$

where ϑ_i^r is defined as the discriminating power of an attribute value A_i^r.

Œ(A_i) finds the association between the attribute A_i and various class decisions the database D has elements of m different classes, and is defined as

$$Œ + (A_i) = \left(\frac{1}{m}\sum_{j=1}^{m} \Lambda_i^j\right) - 1.0 \quad (7)$$

where +(A_i), which denotes the class-to-attribute association for the attribute A_i. The significance of an attribute A_i is computed as the average of Æ(A_i) and Œ (A_i).

B. Wrapper Methods

One wrapper method feature selection, which is Sequential Forward Selection (SFS), is also considered. The best subset of features T is initialized as the empty set and in each step the feature that gives the highest correct classification rate is added to T along with the features already included in T [1]. The process continues until the correct classification rate given by T and each of the features not yet selected does not increase. There are five classifiers used by SFS, which are Naïve Bayes (NBayes), IB1, 1R, Random Forest (RForest), and Negative Selection Algorithm (NSA).

1) NBayes

NBayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with naive independence assumptions [18]. It only requires a small amount of training data to estimate the parameters used for classification. This classifier uses numeric estimator. Numeric estimator precision values are chosen based on analysis of the training data.

2) IB1

IB1 uses normalized Euclidean distance to find the training instance closest to the given test instance, and predicts the same class as this training instance [19]. If multiple instances have the smallest distance to the test instance, the first one found is used.

3) 1R

1R uses the minimum-error attribute for prediction. It ranks attributes according to error rate [20]. It treats all numerically-valued attributes as continuous and uses a straightforward method to divide the range of values into several disjoint intervals. It handles missing values by treating "missing" as a legitimate value. This classifier also discretizes numeric attributes.

4) RForest

RForest constructs a forest of random trees that considers K randomly chosen attributes at each node [21]. This classifier performs no pruning, and also has an option to allow estimation of class probabilities based on a hold-out set.

5) NSA

NSA performs pattern recognition by storing information about the complement set (non-self-cell) to be recognized and provides tolerance for self-cells and deals with the immune system's ability to detect unknown antigens while not reacting to the self-cells [12]. It generates random detectors and eliminates the ones that detect self-cells. Non-self is detected if there is a match between an antigen and any of the detectors.

IV. Experiments

With the goal stated in the section above, an extensive and rigorous empirical comparative study is designed and conducted. In this section, a detailed description of the experimental method is provided.

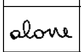
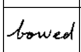
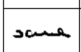
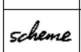
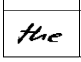
A. Dataset

The comparisons were carried out in dataset coming from the IAM Handwriting Database [22]. IAM Handwriting Database contains forms of handwritten English text which can be used to train and test handwritten text recognizers and to perform writer identification and verification experiments. There are 657 classes available, however only 60 classes are used for experiments. From these classes, 4400 instances are collected. Words from the forms are extracted using United Moment Invariant (UMI) [23] which represents the word features. Table I shows the examples of the data used in this experiment. However, the values shown in Table I are truncated to two-digit precision, while in the experiments, the precision is not truncated, which is up to sixteen-digit precision.

UMI consists of eight real-value features. UMI is commonly used to describe the shape of an image where the scaling, translation, rotation, and reflection or its combinations to the image does not affect the shape features, hence it is invariant [12]. UMI is an extension of Geometric Moment Invariant (GMI), proposed by Hu in 1962.

One of the usages of UMI in machine learning application is handwriting recognition and handwriting identification. However, handwriting recognition deals with the contents conveyed by the image, while handwriting identification tries to differentiate each image to determine the author of those handwritings. Despite that, both of these tasks embark on the same theoretical foundation.

TABLE I. EXAMPLE OF DATA USED IN THE EXPERIMENT

Word	f1	f2	f3	f4	f5	f6	f7	f8
	1.84	1.79	0.91	1.31	0.84	1.00	0.73	1.79
	1.53	1.08	1.12	1.96	0.72	1.49	1.82	1.46
	1.61	1.53	0.53	0.38	0.80	1.26	0.25	3.29
	1.99	8.24	0.65	0.76	3.77	0.20	0.09	2.40
	3.08	2.06	0.52	0.64	0.52	0.82	0.31	2.75

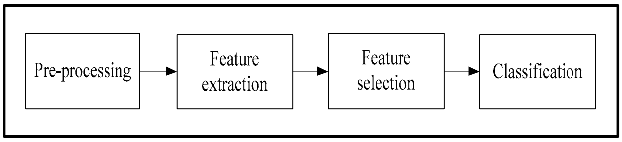


Figure 2. Framework of the experiment

B. Experimental Design

The framework for WI follows the traditional framework of pattern recognition tasks, which are preprocessing, feature extraction, and classification. However, it has been proven that most of preprocessing tasks must be omitted because some of the original and important information are lost, and thus decrease the identification performance in WI domain [12]. Fig. 2 depicts the framework used in the experiment.

UMI is commonly used to determine whether a shape is similar to another shape. However, in this experiment, UMI is not used to find the similar shape; instead it is used to find the similar unique features for the same class (writer). In the previous study, data discretization has been used to improve the classifier accuracy [13], but since this paper focus on the feature selection to improve classifier accuracy, discretization task is omitted and the original data is used.

The three commonly used performance measurements for evaluating the performance of feature selection method are number of selected features, classification accuracy, and processing time. However, this paper only considers two main measures, which are number of selected features and classification accuracy.

As mentioned in the previous section, a total number of 4400 instances are used for the experiments, and are randomly divided into five different datasets to form training and testing dataset in the classification task, as shown in Fig. 3. Every experiment has been performed using ten-fold cross-validation. The result shown is the average of the results produced by each of ten folds. In order to justify the quality of feature subset produced by each method, the feature subsets are tested against classification, which uses NSA as the classifier, and compared against various classifiers as mentioned in Section III.

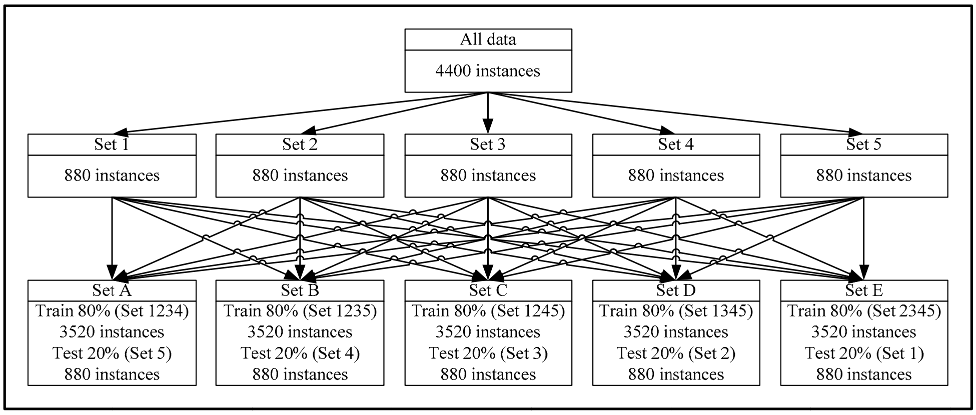


Figure 3. Data collection for training and testing

C. Environmental Setup

This paper uses Waikato Environment for Knowledge Analysis (WEKA) 3.7.1 to measure the performance of each feature selection methods. WEKA came about through the perceived need for a unified workbench that would allow researchers easy access to state-of-the-art techniques in machine learning [24]. WEKA includes algorithms for regression, classification, clustering, association rule mining and attribute (feature) selection. It is also well-suited for developing new machine learning schemes.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

A. Selection Results

The number of features selected by feature selection methods is the primary consideration of this study. Table II shows the number of features selected by each method. Out of ten methods, seven methods have been successfully reduced the number of features to be used.

CFS, FCBF, and SMFS are the algorithms that reduce number of features the most. Both SFS+IB1 and SFS+1R are the second most that reduces the number of feature, followed by LVF and SFS+NSA. Surprisingly, state-of-the-art feature selection algorithm ReliefF is incapable to reduce the number of features at all. It is rather astonishing that two widely used classifiers, NBayes and RForest also are not capable to reduce the number of features, although in set B, SFS+NBayes is capable to reduce the number of features.

ReliefF and SFS+RForest are not capable to reduce the number of features due to the nature of the data itself. ReliefF is known to not be able to detect redundant features. ReliefF use the distance of neighbor samples, so the weight of each feature might have been biased. Meanwhile, SFS+RForest is known prone to over-fitting to some datasets, and thus it is not capable to reduce the number of features. It also does not handle large numbers of irrelevant features.

B. Classification Accurary

The second measurement of this study is classification accuracy, as shown in Table II. The results show that there is no feature selection method yields classification accuracy more than 50%. Even so, previous study conducted in [13] by using discretized and un-discretized data also shows relatively the same classification result. Thus, the classification result not to high have already been expected, although feature selection is added, due to the various shape of words have been used to represent a writer. The results can be improved by integrating discretization, and complete the whole cycle.

Based on the classification results, the accuracy is at its highest when the number of features is between 4-7 features. Overall, LVF yields the most stable and highest classification accuracy, followed by SFS+NBayes, SFS+NSA, and lastly SFS+IB1. As expected from ReliefF and SFS+RForest, the classification accuracy is the same with the original set.

On the other hand, CFS, FCBF, SMFS and SFS+1R produce the classification accuracy which is the lowest among other results. This is because single feature may not have discriminatory power to differentiate each class. However, a combination of at least two features is capable to increase the classification accuracy. SFS+NSA performs quite well in most dataset, although it is not producing stable results. In the contrary, LVF produces the stable results for all datasets.

The average results also show that CFS, FCBF, SMFS, and SFS+1R perform poorly. This is because the feature selection methods are more suitable when handling high-dimensional data, because these methods analyze the correlation between features, which is feature relevancy and feature redundancy. These methods will perform poorly when they failed to find the correlation between features.

TABLE II. EXPERIMENTAL RESULTS

Method	Criteria	Set A	Set B	Set C	Set D	Set E	Average
ReliefF (benchmark)	Number of selected features	8	8	8	8	8	8
	Classification accuracy	45.99%	45.99%	45.99%	45.99%	45.99%	45.99%
CFS	Number of selected features	1	1	1	1	1	1
	Classification accuracy	4.29%	4.29%	4.29%	4.29%	4.29%	4.29%
LVF	Number of selected features	4	4	4	4	4	4
	Classification accuracy	45.65%	45.65%	45.65%	45.65%	45.65%	45.65%
FCBF	Number of selected features	1	1	1	1	1	1
	Classification accuracy	4.29%	4.29%	4.29%	4.29%	4.29%	4.29%
SMFS	Number of selected features	1	1	1	1	1	1
	Classification accuracy	4.29%	4.29%	4.29%	4.29%	4.29%	4.29%
SFS+NBayes	Number of selected features	8	7	8	6	8	7.4
	Classification accuracy	45.99%	48.19%	45.99%	30.76%	45.99%	43.38%
SFS+IB1	Number of selected features	2	2	5	2	2	2.6
	Classification accuracy	30.85%	30.85%	40.93%	30.85%	30.85%	32.87%
SFS+1R	Number of selected features	2	1	1	1	1	1.2
	Classification accuracy	38.75%	4.29%	4.29%	4.29%	4.29%	11.18%
SFS+RForest	Number of selected features	8	8	8	8	8	8
	Classification accuracy	45.99%	45.99%	45.99%	45.99%	45.99%	45.99%
SFS+NSA	Number of selected features	5	4	5	5	4	4.8
(proposed technique)	Classification accuracy	45.76%	48.06%	40.02%	30.21%	40.25%	40.86%

Although ReliefF and SFS+RForest yield the highest average classification accuracy, there are no features reduced and hence the classification result is equals to original set classification result. As discussed in the previous section, the nature of data affects the performance of these two feature selection methods.

Even though there are no feature selection methods capable to increase the classification accuracy more than 50%, however, these results are motivating to develop a novel feature selection method that is able to significantly improve the classification accuracy. This is because even though the classification accuracy does not improved, the number of features is still reduced, and thus reduces the workload of the classification task.

VI. CONCLUSIONS AND FUTURE WORKS

An extensive comparative study on feature selection methods for handwriting identification has been presented. This paper compared the merits of six different feature selection methods; five of them are filter methods, while the last one is wrapper method which uses five different classifiers. The experiments have shown that even though the numbers of features have been reduced, there is no significance improvement toward the classification accuracy.

The wrapper method is confirmed as the best option when it can be applied. In this paper, SFS is selected and used. When wrapper is not applicable, the results suggest using LVF. LVF produces the best results among other methods, both the number of features reduced and the classification accuracy, while some state-of-the-art methods yields poor results, either in number of features reduced, classification accuracy, or both. These results are produced by using commonly used feature selection methods, which is not purposely developed in handwriting identification domain.

Hence, future works to develop a novel feature selection method which is specifically adapted for handwriting identification domain based on this experimental study is required. The proposed feature selection method is going to be compared with feature selection methods discussed in this paper. Both discretized and un-discretized data will be used to perform the comparison; however the study will not be using UMI as the data.

REFERENCES

[1] Dash, M., & Liu, H. (1997). Feature Selection for Classification. *Journal of Intelligent Data Analysis*, 131-156.

[2] Saeys, Y., Inza, I., & Larranaga, P. (2007). A Review of Feature Selection Techniques in Bioinformatics. *Journal of Bioinformatics*, 2507-2517.

[3] Hall, M. A. (1999). Correlation-based Feature Subset Selection for Machine Learning. Hamilton: University of Waikato.

[4] Gadat, S., & Younes, L. (2007). A Stochastic Algorithm for Feature Selection in Pattern Recognition. *Journal of Machine Learning Research*, 509-547.

[5] Portinale, L., & Saitta, L. (2002). Feature Selection: State of the Art. In L. Portinale, & L. Saitta, *Feature Selection* (pp. 1-22). Alessandria: Universita del Piemonte Orientale.

[6] Refaeilzadeh, P., Tang, L., & Liu, H. (2007). On Comparison of Feature Selection Algorithms. *Proceedings of AAAI Workshop on Evaluation Methods for Machine Learning II* (pp. 34-39). Vancouver: AAAI Press.

[7] Zhang, P., Bui, T. D., & Suen, C. Y. (2004). Feature Dimensionality Reduction for the Verification of Handwritten Numerals. *Journal of Pattern Analysis Application*, 296-307.

[8] Kim, G., & Kim, S. (2000). Feature Selection Using Genetic Algorithms for Handwritten Character Recognition. *Seventh International Workshop on Frontiers in Handwriting Recognition* (pp. 103-112). Amsterdam: International Unipen Foundation.

[9] Schlapbach, A., Kilchherr, V., & Bunke, H. (2005). ImprovingWriter Identification by Means of Feature Selection and Extraction. *Eight International Conference on Document Analysis and Recognition* (pp. 131-135). Seoul: IEEE.

[10] Kudo, M., & Sklansky, J. (2000). Comparison of Algorithms that Select Features for Pattern Classifiers. *Journal of Pattern Recognition 33*, 25-41.

[11] Srihari, S. N., Cha, S.-H., Arora, H., & Lee, S. (2002). Individuality of Handwriting. *Journal of Forensic Science*, 856-872.

[12] Muda, A. K. (2009). Authorship Invarianceness for Writer Identification Using Invariant Discretization and Modified Immune Classifier. Johor: Universiti Teknologi Malaysia.

[13] Muda, A. K., Shamsuddin, S. M., & Darus, M. (2008). Invariants Discretization for Individuality Representation in Handwritten Authorship. *2nd International Workshop on Computational Forensics* (pp. 218-228). Washington DC: Springer-Verlag.

[14] Robnik-Sikonja, M., & Kononenko, I. (2003). Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning Journal*, 23-69.

[15] Liu, H., & Setiono, R. (1996). A Probabilistic Approach to Feature Selection - A Filter Solution. *International Conference of Machine Learning* (pp. 319-337). Bari: Morgan Kaufmann.

[16] Yu, L., & Liu, H. (2003). Feature Selection for High-Dimensional Data: A Fast Correlation-Based Filter Solution. *Proceedings of the Twentieth International Conference on Machine Learning* (pp. 856-863). Washington: Machine Learning.

[17] Ahmad, A., & Dey, L. (2004). A Feature Selection Technique for Classificatory Analysis. *Pattern Recognition Letters*, 43-56.

[18] Rish, I. (2001). An Empirical Study of the Naive Bayes Classifier. *Workshop on Empirical Methods in Artificial Intelligence*.

[19] Aha, D., & Kibler, D. (1991). Instance-based Learning Algorithms. *Journal of Machine Learning*, 37-66.

[20] Holte, R. C. (1993). Very Simple Classification Rules Perform Well on Most Commonly Used Datasets. *Journal of Machine Learning*, 63-91.

[21] Breiman, L. (2001). Random Forests. *Journal of Machine Learning*, 5-32.

[22] Marti, U., & Bunke, H. (2002). The IAM-database: an English Sentence Database for Off-line Handwriting Recognition. *International Journal on Document Analysis and Recognition, Volume 5*, 39-46.

[23] Yinan, S., Weijun, L., & Yuechao, W. (2003). United Moment Invariants for Shape Discrimination. *International Conference on Robotics, Intelligent Systems and Signal Processing* (pp. 88-93). Changsha: IEEE.

[24] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA Data Mining Software: An Update. *Journal of SIGKDD Explorations, Volume 11*, 10-18.