An Analysis of Various Algorithms For Text Spam Classification and Clustering Using RapidMiner and Weka

Zainal, K.
Faculty of Science & Technology
Islamic Science University of
Malaysia (USIM)
Nilai, Negeri Sembilan, Malaysia

Sulaiman, N.F.
Faculty of Science & Technology
Islamic Science University of
Malaysia (USIM)
Nilai, Negeri Sembilan, Malaysia

Jali, M.Z.
Faculty of Science & Technology
Islamic Science University of
Malaysia (USIM)
Nilai, Negeri Sembilan, Malaysia

***Abstract*—This paper reported and summarized findings of spam management for Short Message Service (SMS) which consists of classification and clustering of spam using two different tools, namely RapidMiner and Weka. By using the same dataset, which is downloaded from UCI, Machine Learning Repository, various algorithms used in classification and clustering in this simulation has been analysed comparatively. From the simulation, both tools giving the similar results that the same classifiers are the best for SMS spam classification and clustering which are outperformed than other algorithms.**

Keywords-SMS spam; RapidMiner; Weka; Naïve Bayesian (NB); Support Vector Machine (SVM); k-Nearest Neighbour (kNN); K-Mean; Cobweb; Hierarchical clustering; spam classification; spam clustering.

I. INTRODUCTION

Issue of spam has been widely discussed all over the world. Impact caused by spam has been noticing as extremely risky. Nowadays, spam does not only apply to email form but quite numerous to mention such as web spam, SMS spam and instant messaging spam. Spam has been evolved along as the technology advances.

This paper focuses only on SMS text spam, which covers the issue and available technology for spam management. Data mining is the strategy employed as a part of this paper, whereby it is use of automated data analysis techniques to reveal previously undetected relationships among data items. This regularly includes the analysis of data stored in a library or data warehouse. Three of the major data mining techniques are regression, classification and clustering. The heart of spam management basically includes two phases of data mining; classification and clustering of spam.

This paper basically is arranged in sections, as follows. Section 2 summarizes related works in this field; Section 3 explains the methodology applied in this simulation. Section 4 elaborates tools used and the description of the dataset is explained in Section 5. The design of this experiment is justified in Section 6 and the finding of experiment is discussed in the subsequent section together with the conclusion of the simulation.

II. LITERATURE SURVEY OF RELATED WORKS

Technology has been progressively widened in many aspects; from internet to mobile. This technological advance has affect and influences to the people’s lifestyle globally in many ways including communication channels, news release, shopping pattern and much more aspects of daily life. The most outstanding technology now is mobile phone; as the usage of mobile phone has been highly propagate these recent years. Phone calls, text messaging or SMS and accessing Internet are the most common functions of a mobile phone. But almost a couple of decades ago and still, the use of SMS have been definite nuisance to users all over the world because of the misuse of SMS by unscrupulous parties, which is called as spam.

Spam management consists of at least three main phases; classification, clustering and severity determination level, as suggested in Sulaiman et al. [1]. Classification of spam messages as the main process is significantly able in assisting users to differentiate messages between of ham and spam. Identification of spam messages would reduce the possibility of risk since users will simply ignore the message. While as for spam clustering, it is important to find the current trend of spam dissemination. This information is kindly useful to project the possible amount of loss such as phishing contents might have a higher impact of risk comparing to message with free ringtone offer.

Many researches have been done and it is still an ongoing process in developing filtering spam. Mahmoud et al. [2] has developed a framework to filter SMS spam using a novel introduced algorithm that is mimic human body defence systems knowingly as Artificial Immune System (AIS). In their analysis, findings showed that their proposed engine is giving 91% of accuracy rate compared the performance of Naïve Bayesian (NB) with only 88%.

This research is funded by RAGS, under Ministry of Education, Malaysia.

Rafique et al. [3] proved that Support Vector Machine (SVM) is giving 93%, the highest accurate value in their developed spam filtering framework, compared with other algorithms such as NB, C4.5 and Repeated Incremental Pruning to Produce Error Production (RIPPER).

Cai et al. [4] had developed a system for spam detection using Winnow algorithm. This experiment particularly executed with Chinese language SMS messages. Although the experimental results illustrated that this system works well, it is possible that the system could be further enhanced by developing the feature selection method and the decision making procedure.

III. Methodology: Supervised And Unsupervised Machine Learning Algorithms

Classification of spam is referring to process of messages differentiation between spam and ham (valid or non-spam messages), while for clustering, it is a process of partitioning messages into cluster or group according to its similarity features or characteristics [5].

Both processes can be accomplished with the aid of machine learning with the establishment of various algorithms. Machine learning algorithms are structured into taxonomy that is based on the desired outcome. It is a subfield from the broad field of artificial intelligence, which intend to make machines be able to learn like human [6]. Common algorithm used include supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning, transduction and learning to learn machine learning [7].

With reference to Donalek [5], classification process is implemented using supervised machine learning, while clustering applying unsupervised machine learning. As this experiment involved a process of classification and clustering spam messages, hence both supervised and unsupervised machine learning are applied.

Algorithms that are chosen to be applied in tools of RapidMiner and Weka are elaborated in the following paragraphs. All of these six chosen algorithms are prominent for its performance in data mining field and available in both tools.

A. Classification Implemented Using Supervised Machine Learning Algorithms

According to Brownlee [8], supervised learning deploy an input data that is called training data and has a pre-defined label or result, for example spam / not spam or a stock price at a time. A model is arranged through a training process where it is obliged to make predictions and is corrected when those predictions are wrong. The training process continues until the model achieves a desired level of accuracy on the training data. Example problems that apply supervised learning are classification and regression. The following highlights on various supervised learning algorithm used in this simulation.

1) Naïve Bayesian

Naïve Bayesian (NB) classifier is a simple probabilistic classifier based on concerning Bayes's hypothesis with strong independence theory. A more descriptive term for the underlying probability model would be an 'independent feature model'. The NB inducer computes conditional probabilities of the classes given the instance and picks the class with the highest posterior. Depending on the precise nature of the probability model, NB classifier can be trained very efficiently in a supervised learning setting [9].

According to Awad et al. [6], NB classifier was proposed for spam recognition in 1998. In NB, word probabilities play the main rule which every word has certain possibility of occurring in spam or ham messages in its database. If some words occur often in spam but not in ham, then an incoming message is probably a spam.

2) Support Vector Machine

Support Vector Machine (SVM) is a learning algorithm with 2-class classification method. This algorithm converts miscellaneous domain knowledge with overlapping inputs into non-overlapping parametric objects by modelling the instances from the input space to the feature space using kernel functions. The classification is done by constructing a hyper plane between instances of different classes [3].

According to El-halees [10], *y* which classifies messages as spam or legitimate according to following dot product:

$$y = w.x - b \tag{1}$$

where *x* is a feature vector of messages composed of words. *w* is the weight of corresponding *x*. *b* is a bias parameter determined by training process.

3) k-Nearest Neighbour

k-Nearest Neighbour (kNN) classifier is considered as an example-based classifier, means the training documents are used for comparison, rather than an explicit category representation, such as the category profiles used by other classifiers. As such, there is no real training phase. When a new document need to be categorized, the *k* most similar documents (neighbours) are found and if a large enough proportion of them have been assigned to a certain category, the new document is also assigned to this category, otherwise not. Additionally, finding the nearest neighbours can be accelerated using traditional indexing methods. To decide whether a message is spam or ham, it is referring to the class of the messages that are closest to it. The comparison between the vectors is a real time process. This is the idea of kNN algorithm [6].

B. Clustering Implemented Using Unsupervised Machine Learning Algorithms

Brownlee [8] defined unsupervised learning has an input data that is not labelled and does not have a known result. A model is prepared by deducing structures present in the input data and example problems are association rule learning and clustering. The following highlights on various unsupervised machine learning algorithm used in this simulation.

1) K-Means

K-Means clustering algorithm was first proposed by Macqueen in 1967 which was uncomplicated, non-supervised learning clustering algorithm. K-means is a partitioning clustering algorithm, this technique is used to classify the given data objects into *k* different clusters through iterative method, which tends to converge to a local minimum. So, the outcomes of generated clusters are dense and independent of each other [11].

The K-Means algorithm is the best known squared error-based clustering algorithm [12]. It involves the processes of:

- Selection of the initial *k* means for *k* clusters;
- Calculation of the dissimilarity between an object and the mean of a cluster;
- Allocation of an object to the cluster whose mean is nearest to the object; and
- Re-calculation of the mean of a cluster from the objects allocated to it so that the intra cluster dissimilarity is minimized.

2) Cobweb

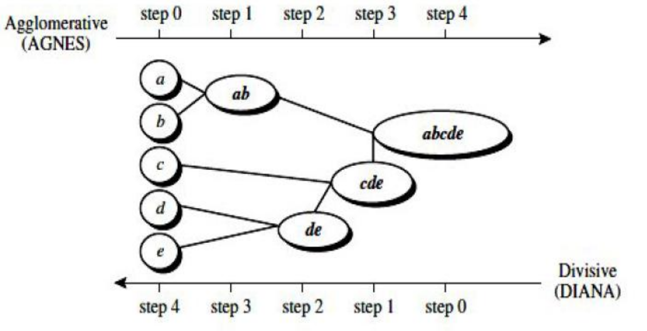
The Cobweb algorithm was developed by machine learning researchers in 1980s for clustering objects in an object-attribute dataset. This algorithm yields a clustering dendrogram called classification tree that differentiates each cluster with a probabilistic description. Cobweb generates hierarchical clustering where clusters are described probabilistically [13].

3) Hierarchical

Hierarchical method creates a hierarchical decomposition of the given set of data objects forming a dendrogram - a tree which splits the database recursively into smaller subsets. The dendrogram can be formed in two ways, bottom up or top down. Hierarchical algorithm combines or divides existing groups, creating a hierarchical structure that reflects the order in which groups are merged or divided.

The bottom up approach, also called the agglomerative approach, starts with each object forming a separate group. It successively merges the objects or groups according to some measures like the distance between two centres of two groups and this is done until all of the groups are merged into one, or until a termination condition holds.

The top down also called the divisive approach, starts with all the objects in the same cluster. In each successive iteration, a cluster is split into smaller clusters accordingly to some measures until eventually each object is in one cluster, or until a termination condition holds.



Agglomerative and divisive hierarchical clustering on data objects $\{a, b, c, d, e\}$.

Figure 1. Hierarchical clustering process [14].

IV. TOOLS

These whole activities of spam classification and clustering can be done with the aid of established algorithms that has been developed using software. These tools are available as commercial or free products.

As for the implementation of this experiment, two tools with free license (freeware) has been used in this simulation, RapidMiner (Community Edition) and Weka. Both tools are prominent as data mining software and can be downloaded from the Internet.

Christa et al. [15] in their paper evaluated and analyzed comparatively of various data mining tools such as KNIME, RapidMiner, Weka, Tanagra and Orange. These tools has its own advantages and unique. Among all these data mining tools, Weka and RapidMiner have the biggest and most active user communities. Both of them quickly implement (and integrate) new and emerging machine learning algorithms into their systems.

As in this paper, the simulation process is using RapidMiner and Weka as the data mining tools.

A. RapidMiner

RapidMiner is data mining software, which can be use as a standalone application for data analysis or integrate as a data-mining engine into other products. This tool has unique features such as:

- Data integration, analytical Extract Transform Load (ETL), data analysis and reporting into a single suite;
- Powerful intuitive Graphical User Interface (GUI) for the design of analytical processes;
- Repository for process, data and metadata management;
- Metadata transformation which results inspection available during design;
- Support on-the-fly error detection and quick fixes; and
- Complete and flexible with hundreds of methods available for data integration, data transformation, modelling and visualization.

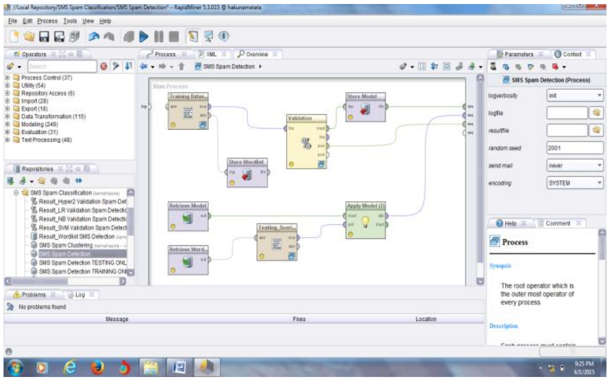
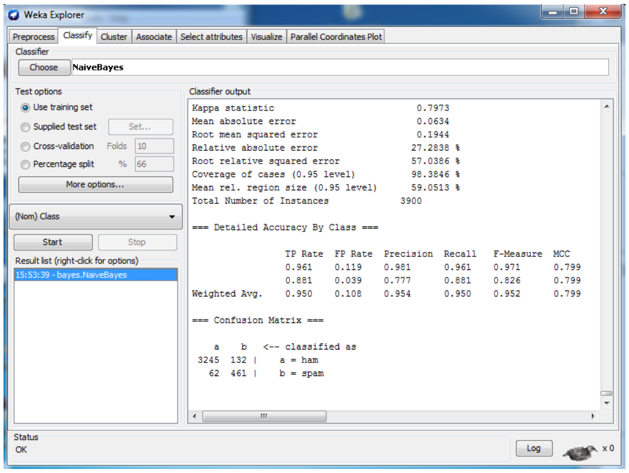


Figure 2. Interface of RapidMiner.

B. Weka

Weka is a collection of machine learning algorithms for data mining tasks with GUI. This application is named after a flightless bird of New Zealand that is very inquisitive. The algorithms can either be applied directly to a dataset or called from own Java code. Weka contains feature for data pre-processing, classification, regression, clustering, association rules and visualization. It is also well-suited for developing new machine learning schemes. There are four buttons of GUI in Weka [16] which are:

- Explorer – an environment for exploring data
- Experimenter – an environment for performing experiments and conducting statistical tests between learning schemes
- Knowledge Flow – this environment supports essentially the same functions as the Explorer but with a drag and drop interface and it supports incremental learning
- Simple CLI – provides a simple command line interface that allows direct execution of Weka commands.



Classifier: NaiveBayes

Test options:
● Use training set
○ Supplied test set
○ Cross-validation Folds 10
○ Percentage split % 66

(Nom) Class

15:53:39 - bayes.NaiveBayes

```
Kappa statistic                          0.7973
Mean absolute error                      0.0634
Root mean squared error                  0.1944
Relative absolute error                 27.2838 %
Root relative squared error             57.0386 %
Coverage of cases (0.95 level)          98.3846 %
Mean rel. region size (0.95 level)      59.0513 %
Total Number of Instances             3900

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC
                 0.961    0.119    0.981      0.961    0.971      0.799
                 0.881    0.039    0.777      0.881    0.826      0.799
Weighted Avg.    0.950    0.108    0.954      0.950    0.952      0.799

=== Confusion Matrix ===

    a    b   <-- classified as
 3245  132 |    a = ham
   62  461 |    b = spam
```

Status: OK

Figure 3. Interface of Weka.

Both of the aforementioned tools are deployed in this simulation by an application of specified algorithms that has been stated earlier.

As to summarize this experiment, this simulation is depicted in the following figure that exhibit the integration between spam management processes, machine learning algorithms and data mining tools; RapidMiner and Weka.

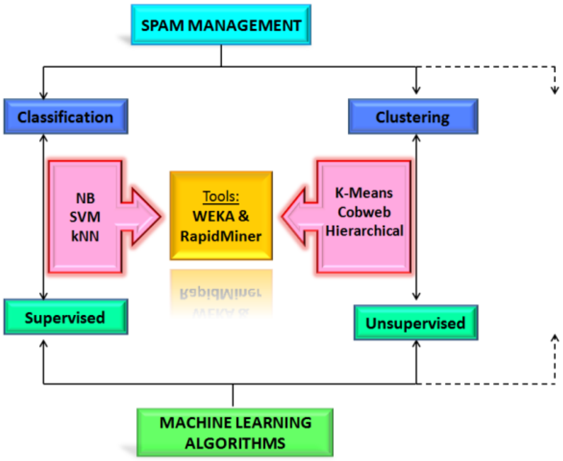


Figure 4. Integration of spam management processes with machine learning algorithms and experimental tools.

V. Dataset Description

This experiment is using a dataset downloaded from UCI, Machine Learning Repository. This corpus site stored a collection of public set of SMS messages labelled as spam and ham (non-spam) for the use by many researchers. The dataset downloaded consists of 5,572 instances which is 4,825 messages are labelled as ham and 747 as spam.

All 5,572 downloaded SMS messages from this corpus are used both for training and testing phases with the fraction of 7:3 (7 for training and 3 for testing). 70% of both ham and spam messages are used in training phase, as the more the dataset use for training, the better the model would be when it is applied in the testing phase. The other 30% of both ham and spam messages are used in testing phase. This dataset of SMS messages are divided as follows:

TABLE I. ALLOCATION OF THE DATASET INTO TWO PARTS, TRAINING AND TESTING PROCESS

	Training phase (70%)	Testing phase (30%)	Total messages
Labelled as HAM	3,377	1,448	4,825
Labelled as SPAM	523	224	747
Total	**3,900**	**1,672**	**5,572**

As to provide a fair comparison, this same dataset is deployed using six aforementioned algorithms in RapidMiner and Weka.

VI. EXPERIMENTAL DESIGN

As stated earlier, spam management involved three main processes, which are spam classification, clustering and severity determination of the detected spam. However, the focal objective of this experiment is focusing on the first two processes; namely classification and clustering. Different classifiers or algorithms models are applied in RapidMiner and Weka (only used Explorer as GUI application), purposely to find the best suited algorithm for those two processes.

While as for the data mining tools, this experiment is using RapidMiner version 5.3.015 and Weka version 3.7.10.

A. Spam Classification

As for the classification process, there are two levels that applied: training and testing level. These two different levels are actually reflected the supervised machine learning characteristics.

During training, a set of 3,377 ham and 523 spam labelled messages are running through a few different classifiers separately. Then, for testing, based on stored correlation attributes during the training level, a set of unlabelled messages are running through those classifiers. Finally the results of these findings are verified to measure its performance.

In spam classification process, there are four methodologies that the testing of unlabelled messages has been executed, as explain in Table II. All classifiers had been re-run in Method 1 and 2, while Method 3 and 4 only re-run using the best classifier identified in Method 1 and 2, since the result of best classifier in Method 3 and 4 remain the same as in Method 1 and 2.

TABLE II. DESCRIPTION OF EXECUTION PLAN FOR TRAINING AND TESTING PHASE

Method	**No. of messages**		**Description of testing phase**
	Training phase	**Testing phase**	
1	3,900 messages labelled with ham or spam	1,672 unlabelled messages	These 2 phases of spam detection are run simultaneously. The time taken to complete the whole process is recorded.
2	3,900 messages labelled with ham or spam	1,672 unlabelled messages	These 2 phases of spam detection are run separately. The time taken to complete these 2 different processes is recorded.
3	3,900 messages labelled with ham or spam	1 unlabelled message	These 2 phases of spam detection are run separately. As for the testing phase, a few different unlabelled messages are run repeatedly, with only 1 unlabelled message run at one time (reflect the actual environment).
4	100 messages labelled with ham and spam	1,672 unlabelled messages	These 2 phases of spam detection are run separately. As for the testing phase, 1,672 unlabelled messages are run repeatedly with a different size of database stored during training phase. This method is to demonstrate the link between spam library / database (developed during training) with unlabelled messages run in testing phase. Also to find the influence degree of number of messages used in training with the result of spam classification, in term of accuracy rate.
	1,250 messages labelled with ham and spam		
	2,500 messages labelled with ham and spam		

B. Spam Clustering

Clustering is concerned with grouping together SMS spam messages that are similar to each other and dissimilar to the other clusters or groups. Clustering is a technique for extracting information from unlabelled data. This is important as to learn the pattern of spam content.

At this phase, all SMS messages that have been pre-classified as spam are used for further process, to be cluster according to its category by referring to the content of messages. Since clustering process is employing an unsupervised machine learning algorithm, there is not a requirement to run a training dataset. Therefore, all 747 spam messages from the dataset are used to be further tested with different classifiers. Referring to Delany et al. [17], these 747 spam messages have been categorized into 10 clearly defined groups, which are:

- Competitions
- Chat
- Claims
- Dating
- Prizes
- Services
- Finance
- Ringtones
- Voicemail
- Miscellaneous

VII. Performance Measurement

In order to rank or select the best classifier in SMS spam classification, some of the performance measurements need to be choosing to determine the best option.

As to measure the best performance, the higher the value of Accuracy, the better the classifier it is. The time taken to complete the process also selected as one of the criteria in performance measurement. The shorter the time taken, the better the classifier it is. The following measurements are used to formulate the performance that commonly used in machine learning [18].

- Accuracy in percent (%): the value of spam being correctly classified. It is also reflect the overall performance of the framework.

$$A = \frac{TP+TN}{TP+TN+FP+FN} \qquad (2)$$

- True Positive (TP): the number of SMS spam classified as spam.
- True Negative (TN): the number of SMS ham classified as ham.
- False Positive (FP): the number of SMS ham falsely classified as spam.
- False Negative (FN): the number of SMS spam falsely classified as ham.

Processing time in seconds: the time taken to complete the process. The shorter the time taken the better the classifier it is.

TABLE III. Confusion Matrix Of Classification Algorithm

		Prediction	
		Spam	Ham
True	Spam	**TP**	**FN**
	Ham	**FP**	**TN**

VIII. Results and Discussion

A. Spam Classification

These 1,672 instances of dataset have constructed 5,209 regular attributes in RapidMiner. The result of SMS spam classification for unlabelled 1,672 messages (1,448 ham and 224 spam messages) that running through with different classifiers in both tools are table out as below:

Method 1:

TABLE IV. Accuracy Rate and Time Taken To Process Messages (Training And Testing Phases Executed Simultaneously)

	Accuracy (%)		**Processing time (seconds)**	
	RapidMiner	**Weka**	**RapidMiner**	**Weka**
NB	84.79	94.56	279	0.91
SVM	96.64	98.21	263	2.48
kNN	94.74	94.80	802	0.01

Method 2:

TABLE V. Accuracy Rate And Time Taken To Process Messages (Training And Testing Phases Executed Separately)

	Accuracy (%)		**Processing time (seconds)**			
	Rapid Miner	**Weka**	**RapidMiner**		**Weka**	
			Training	**Testing**	**Training**	**Testing ****
NB	84.79	95.03	173	38	0.64	-
SVM	96.64	99.33*	195	21	1.54	-
kNN	94.74	99.85	1091	493	6.0	-

*Even though kNN is giving a slightly higher accuracy rate compared to SVM in Weka, time taken to process messages is still significantly lesser than SVM.
**means there is no record of time taken for testing phase in Weka.

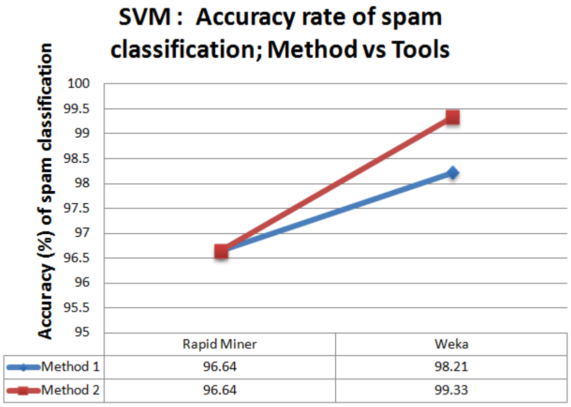


Figure 5. Accuracy rate of spam classification for Method 1 and Method 2 using RapidMiner and Weka.

As for spam classification, it showed that SVM is the best classifier in both RapidMiner and Weka.With referring to graphical representation in Figure 5, RapidMiner giving the same accuracy rate (96.64%) of spam classification in both methods (training and testing phases executed simultaneously and separately), but different in processing time. On the other hand, deployment of Method 2 in Weka is giving a slightly higher of accuracy rate (99.33%) in spam classification compared to Method 1 (98.21%).

Method 3:

TABLE VI. COMPARISON OF PREDICTIVE RESULTS AND TRUE LABEL OF UNLABELLED MESSAGES USING SVM CLASSIFIER IN RAPIDMINER

	Processing time (seconds)		**Description of findings in testing phase**		
	Training phase	**Testing phase**	**Unlabelled message**	**Prediction by classifier**	**True label**
SVM	195	0	202.txt	Ham	Ham
			567.txt	Ham	Ham
			862.txt	Ham	Ham
			1198.txt	Ham	Ham
			1443.txt	Ham	Ham
			1578.txt	Spam	Spam
			1599.txt	Spam	Spam
			1616.txt	Spam	Spam
			1658.txt	Spam	Spam
			1670.txt	Spam	Spam

TABLE VII. COMPARISON OF PREDICTIVE RESULTS AND TRUE LABEL OF UNLABELLED MESSAGES USING SVM CLASSIFIER IN WEKA

	Processing time (seconds)		**Description of findings in testing phase**		
	Training phase	**Testing phase**	**Unlabelled message**	**Prediction by classifier**	**True label**
SVM	1.51	-	One.arff	Ham	Ham
			Two.arff	Ham	Ham
			Three.arff	Ham	Ham
			Four.arff	Ham	Ham
			Five.arff	Ham	Ham
			Six.arff	Spam	Spam
			Seven.arff	Spam	Spam
			Eight.arff	Spam	Spam
			Nine.arff	Spam	Spam
			Ten.arff	Spam	Spam

Messages are in separate text file and randomly choose to be further tested. Results show 100% accurate, whereby prediction by classifier are match with the actual label of the messages.

Method 4:

TABLE VIII. RESULTS OF TESTING PHASE (ACCURACY AND TIME TAKEN) WHEN VARIOUS DATABASE ARE DEPLOYED USING SVM IN RAPIDMINER

	No. of messages used in training phase	**Processing time (seconds)**		**Accuracy (%)**
		Training	**Testing**	
SVM	100	15	7	65
	1,250	69	14	93.28
	2,500	138	16	94.72
	3,900	195	21	96.64

Referring to Table I, as for the unlabelled messages in testing phase, 1,448 messages is ham and 224 messages is spam.

TABLE IX. RESULT OF TESTING PHASE (ACCURACY AND TIME TAKEN) WHEN VARIOUS DATABASE SIZE ARE DEPLOYED USING SVM IN WEKA

	No of messages used in training phase	**Processing time (seconds)**		**Accuracy (%)**
		Training	**Testing**	
SVM	100	0.02	-	99.0
	1,250	0.25	-	94.96
	2,500	0.48	-	96.08
	3,900	1.54	-	99.33

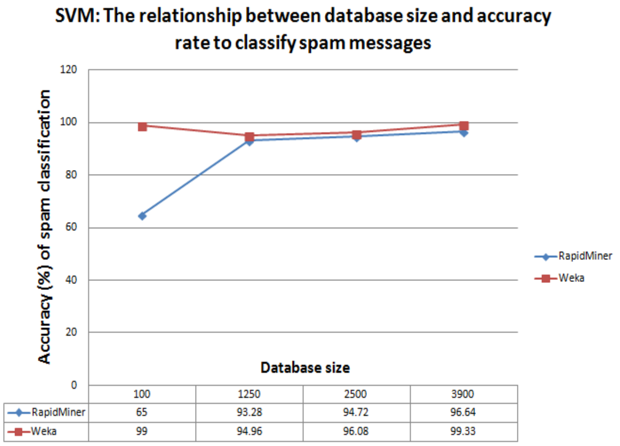


Figure 6. The relationship between database size and accuracy rate of spam classification using RapidMiner and Weka (Method 4).

As referring to Figure 6, this experiment showed that the more the labelled messages are deployed during the training phase as to establish the database library, the better the classifier would become. This result also showed again that Weka is giving a slightly higher accuracy rate compared to RapidMiner.

B. Spam Clustering

As explained in paragraph 6.2, there are 10 groups of spam messages that have been pre-defined of its category. Hence, as for the number of cluster to be defined in every classifier is chosen as 10.

TABLE X. CLUSTERING OF 747 SPAM MESSAGES USING 3 DIFFERENT ALGORITHMS IN RAPIDMINER AND WEKA

	Results of prediction		Processing time (seconds)	
	RapidMiner	**Weka**	**RapidMiner**	**Weka**
K-Means	C0: 343 items C1: 47 items C2: 83 items C3: 17 items C4: 50 items C5: 20 items C6: 39 items C7: 33 items C8: 66 items C9: 49 items	C0: 212 items C1: 2 item C2: 15 items C3: 2 items C4: 6 items C5: 2 items C6: 501 items C7: 2 items C8: 2 items C9: 3 items	37.0	2.7
Hierarchical	Unable to cluster messages.	C0: 729 items C1: 2 items C2: 2 items C3: 2 items C4: 2 items C5: 2 items C6: 2 items C7: 2 items C8: 2 items C9: 2 items	-	0.94
Cobweb	C0:747 items	C0:747 items	31.0	11.22

*C = Cluster

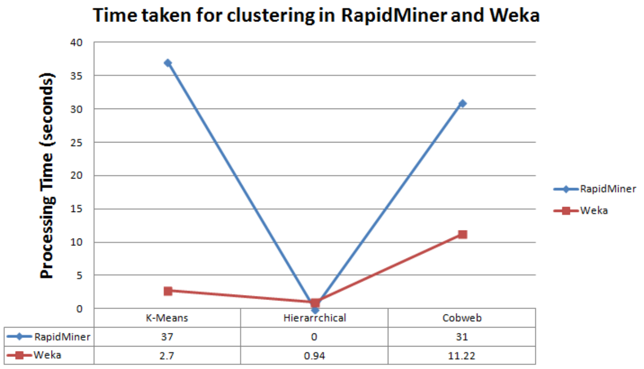


Figure 7. Time taken (seconds) for clustering using K-Means, Hierarchical and Cobweb; in RapidMiner and Weka.

The findings suggest that, with the use of RapidMiner and Weka in SMS spam classification and clustering, it showed that:

- SVM classifier is the best to be used in SMS spam classification, which the result of testing for 1,672 unlabelled messages produced in 21 seconds using RapidMiner with 96.64% of accuracy and 1.54 seconds with 99.33% of accuracy using Weka; and
- K-Means algorithm is the best suited to cluster 747 spam messages into 10 groups in 37 seconds using RapidMiner and 2.7 seconds using Weka.

IX. CONCLUSION

The summary of the comparison in spam classification and clustering using RapidMiner and Weka are tabulated in Table XI.

TABLE XI. THE COMPARISON BETWEEN RAPIDMINER AND WEKA

Phase of Spam Management	WEKA		RapidMiner	
	Best classifier	**Accuracy & Time taken**	**Best classifier**	**Accuracy & Time taken**
Spam Classification	SVM	Accuracy: 99.33% Time taken: 1.54 seconds.	SVM	Accuracy: 96.64% Time taken: 21 seconds for testing phase only for separate execution.
Spam Clustering	K-Means	Time taken: 2.7 seconds.	K-Means	Time taken: 37.0 seconds.

According to the XI, this experiment demonstrated that SVM is the best classifier for spam classification and K-Means is the most suitable algorithms to cluster spam messages. These algorithms giving a promising result both in RapidMiner and Weka. Other than that, this experiment also shows:

- As elaborated in Method 2 findings, it suggest that the testing phase will take shorter time when it is run separately with training phase;
- As the size of datasets increases, time taken to classify the messages are also increase;
- Logically, in practice only one or two SMS will be delivered to mobile phone at one time, and this will not consume much time in classification process to detect either it is a spam or not, as suggest in the findings of Method 3;
- As suggest in Method 4, the more the dataset used in training phase, the better the classifier will perform and giving a higher accuracy rate but as the size of datasets increases, the time taken to 'learn' and classify the spam messages also increases in both training and testing phases;
- Weka tool is giving the shortest time in executing the spam classification and clustering and also giving a higher rate of accuracy which is significantly better compared to RapidMiner;
- Both tools resulted the same sequence of highest accuracy in spam classification; SVM, KNN and NB;
- Cobweb is not a suitable algorithm to cluster SMS messages. All 747 spam messages were clustered into one group only instead of 10 groups, both in RapidMiner and Weka; and
- This study revealed that the same classifier performed dissimilarly when run on the same dataset but using different tools.

The main motivation for different classification algorithms is resulting in high accuracy rate. Each method has its own variety of algorithms. Various algorithms of these methods were used to predict the pattern and behaviour of the dataset, as in this case is spam messages.

This similar simulation can be executed for other different and advanced algorithm. An employment of Artificial Immune System (AIS) is one of the techniques that can be further considered since this AIS has been well developed and matured to be tested in many field to detect malicious behavior such as email spam classification, virus and intrusion detection.

ACKNOWLEDGMENT

Authors wish to thank the Ministry of Education, Malaysia for funding this research. The fund is known as RAGS, under the code number RAGS/2013/USIM/ICT04/2.

REFERENCES

[1] Sulaiman, N. F., and Jali, M. Z. "Integrated Mobile Spam Model Using Artificial Immune System Algorithms", Knowledge Management International Conference (KMICe), 2014.

[2] Mahmoud, T. M., and Mahfouz, A. M. "SMS Spam Filtering Technique Based on Artificial Immune System", International Journal of Computer Science, 2012.

[3] Rafique, M. Z., Alrayes, N., and Khan, M. K. 2011. Application of Evolutionary Algorithms in Detecting SMS Spam at Access Layer.

[4] Cai, J., Tang, Y., and Hu, R. "Spam Filter for Short Messages using Winnow", International Conference on Advanced Language Processing and Web Information Technology, 2008.

[5] Donalek, C. 2011. Supervised and Unsupervised Learning.

[6] Awad, W. A., and ELseuofi, S. M. "Machine Learning Methods for Spam Email Classification", International Journal of Computer Science and Information Technology, 2011.

[7] Ayodele, T. O. 2010. Types of Machine Learning Algorithms. In New Advances in Machine Learning.

[8] Brownlee, J. 2013. A Tour of Machine Learning Algorithms. http://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/.

[9] Lakshmi, R. D., and Radha, N. "Supervised Learning Approach for Spam Classification Analysis using Data Mining Tools", International Journal on Computer Science and Engineering, 2010.

[10] El-halees, A. "Filtering Spam E-Mail from Mixed Arabic and English Messages: A Comparison of Machine Learning Techniques", International Arab Journal of Information Technology, 2009.

[11] Sehgal, G., and Garg, D. K. "Comparison of Various Clustering Algorithms", International Journal of Computer Science and Information Technologies, 2014.

[12] Saxena, P., and Lehri, S. "Analysis of Various Clustering Algorithms of Data Mining on Health Informatics", International Journal of Computer and Communication Technology, 2013.

[13] Sharma, N., Bajpai, A., and Litoriya, R. " Comparison the various clustering algorithms of weka tools", International Journal of Emerging Technology and Advanced Engineering, 2012.

[14] Godara, S. "A Comparative Performance Analysis of Clustering Algorithms", International Journal of Engineering Research and Applications.

[15] Christa, S., Madhuri, K., and Suma, V. 2012. A Comparative Analysis of Data Mining Tools in Agent Based Systems.

[16] Chaudhari, B., and Parikh, M. " A Comparative Study of Clustering Algorithms Using Weka tools", International Journal of Application or Innovation in Engineering and Management (IJAIEM), 2012.

[17] Delany, S. J., Buckley, M., and Greene, D. 2012. SMS Spam Filtering: Methods and Data. Expert Systems with Applications , Elsevier, 01(10).

[18] Wang, A. H. 2012. Machine Learning for the Detection of Spam in Twitter Networks. *ICETE*, 319–333.

AUTHORS PROFILE

Kamahazira Zainal currently is pursuing her Ph.D in Science and Technology specifically in Information Security and Assurance from Islamic Science University of Malaysia (USIM). Previously received her Master in Information Security from Universiti Teknologi Malaysia (UTM) in 2008 and Bachelor of Computer and Communication Engineering from Universiti Putra Malaysia (UPM) in year of 2000.

Nurul Fadhilah Sulaiman presently studying Master in Information Security from Islamic Science University of Malaysia (USIM). She received her degree in Computer Science (Information Security and Assurance) in 2013, also from USIM. She already published a paper related to this field titled "Integrated Mobile Spam Model Using Artificial Immune System Algorithms" in Knowledge Management International Conference 2014 (KMICe 2014).

Dr. Mohd Zalisham Jali received his Ph.D in 2011 from the Plymouth University, UK. Dr Zalisham is now senior lecturer of the Computer Science program. His current research of interest includes information security, web accessibility and e-learning.