Evaluation of the Performance for Popular Three Classifiers on Spam Email without using FS methods

Ghada AL-Rawashdeh, Rabiei Bin Mamat
Ocean engineering Technology and Informatics Dept
Universiti Malaysia Terengganu
MALAYSIA

Jawad Hammad Rawashdeh
Computing and Informatics Dept
Saudi Electronic University
SAUDI ARABIA

Abstract: - Email is one of the most economical and fast communication means in recent years; however, there has been a high increase in the rate of spam emails in recent times due to the increased number of email users. Emails are mainly classified into spam and non-spam categories using data mining classification techniques. This paper provides a description and comparative for the evaluation of effective classifiers using three algorithms - namely k-nearest neighbor, Naive Bayesian, and support vector machine. Seven spam email datasets were used to conducted experiment in the MATLAB environment without using any feature selection method. The simulation results showed SVM classifier to achieve a better classification accuracy compared to the K-NN and NB.

Key-Words: - Classification Accuracy; KNN; SVM; NB; Email Spam

1 Introduction

Spam cannot be specifically defined as most of them are considered as unsolicited e-mail; however, not all unsolicited e-mails are spam, they could be said to be unsolicited commercial e-mail [1];[35], but most advertising materials are not spam as well [2][34]. Sometimes, spams are called junk emails but then, the question, what is a junk mail? Even though most e-mail users can recognize spam emails, yet, there is no definite definition of spam and spam. There are two categories of text mining - text classifiers and text clustering. Text classification is a supervised learning task which essentially does not need pre-determined documents labels or categories. Its major aim is the detection of new events based on certain criteria. Text classification approach is divided into training and testing phases. The training phase involves the use of documents known as training sets to build the classifier through the assignment of a training subset for each category before using several information retrieval methods processing them and extract the characterizing features for each category. The remaining documents, called test set, are used in the testing phase to evaluate the classifiers' performance via the classification of the documents in each category as unseen documents and measuring the classification performance by comparing the estimated categories to the pre-defined ones. A text can be represented as a set of features using two representation methods; these are Bag-Of Word (BOW) which involves the use of single words or phrases as features, and n-gram which involves using sequence of words (WordLevel n-gram) or characters (Character Level n-gram) of the length n [3].

The handling of the huge number of features (sometimes in the orders of tens of thousands) is the major problem of the building TC system [4-6]. Many IR techniques have been deployed for feature space dimensions, such as Stemming, Feature Selection (FS), and Stop-words Removal. FS techniques such as Odds Ratio (OR), GSS Coefficient (GSS), Mutual Information (MI), Chi-Square Statistic (CHI), and Information Gain (IG) and are deployed for feature space dimensionality reduction and are considered irrelevant for a specific category [7-9]; [10].[35].

The supervised algorithms require a set of labeled documents (since they assume a pre-knowledge of the structure of the text category in a database) in order to accurately map documents to their pre-defined class labels. Earlier, it was discussed that a pre-knowledge of the category structure and the generation of the correctly labeled training set are tedious tasks which are almost impossible in large text databases. Some of the common supervised algorithms are Naive Bayes, k-NN, and SVM [11-16].

Different spam classifiers have been classified by [17] using "bag of word" as the extraction technique but without FS. The classification result show NB to perform better than SVM and tree-based J48. Furthermore, NB, tree-based J48, and IB1 have been classified by [18] using "bag of word" as extraction and the outcome showed NB as the best classifier compared to Pearson correlation, Mutual Information, Chi-square, and Symmetric Uncertainty. However, Chi-square performed a better feature selection using IB1 as a classifier compared to the others.[19] compared SVMs, AdaBoost, and Random Forests (RF) using "bag of word" as an extraction technique and found SVM as the best classifier. The performance of Information Gain and Chi-Square was also comparable to that of SVM while using Chi-Square and RF (for feature selection and as classification method respectively) performed better than using SVM with FS. [20] used six FS methods with SVM and NB on six datasets. They produced a hybrid method (HBM) by combining the optimal document Frequency-based feature selection (ODFFS) with term frequency-based feature selections (TFFSs) and proposed parameter optimization using feature subset evaluating parameter optimization (FSEPO). The performance of NB was reported to be better than that of SVM; the proposed HBM enhanced the process of finding the optimal features for feature selection. This work motivated the use of three best classifiers in this paper to select the best one in the performance during spam classifier.

Machine learning techniques presently used to filter spam e-mail at a highly successful rate. this work proposed for improving the identification of cruel spam in email.so, we use the three classifiers identified in spam detection: Support Vector Machine, Naïve Bayes, and KNN by measuring the performance of the classifiers to know which the best between them and also the classifier algorithms typically use a bag of words features to identify spam e-mail, which an approach commonly used in text classification.

2 Research Method

The supervised algorithms assume a pre-knowledge of the category of text structure in a database; hence, they require a set of labeled documents to correctly map documents to their pre-determined labels. A pre-knowledge of the category structure and the generation of the correctly labeled training set are tedious tasks which are almost impossible in large text databases. This section discussed k-NN, NB, and SVM which are the most popular supervised algorithms evaluated in this study.

2.1 K-Nearest Neighbor Classification

The K-NN is a well-known learning method (instance-based) which has demonstrated strong text categorization performances [21]; [22] It is based on the following principles: First, assume x as a given test document; the k nearest neighbors among the training documents are found and the category of the test document is determined using the category labels of these neighbors. The conventional approach assigns the commonest category label among the k nearest neighbors to the document.

However, the weighted k-NN is an extension of the conventional approach in which the similarity each k nearest neighbor to the test document x is used to weight its contribution. Then, the category score for x is obtained by summing the similarity of the neighbors in each category. This implies that

the score of categories cj for the test document x is:

$$\text{Score (cj,x)} = \sum_{d_i \in N(x)} \cos(\text{x}, \text{d}_i) \,.\, \text{y}(\text{d}_i, \text{c}_j) \quad (1)$$

where N(x) = set of k training documents nearest to x; di = the training document; cos (x, di) = cosine similarity the training document di and between the test document x, and y (di, cj) = a function whose value is 1 if di is in category cj and 0 if not. x is assigned to the highest-ranking category.

Four classifiers (RF, NB, SVM, and K-NN, with BOW as extraction) have been used in a study by[24] which showed RF to perform better than K-NN and the other classifiers.

- BEGIN: Suppose k is the number of nearest neighbors, D is the training set
- For each test instance r= (x', y') do
- Calculation of the distance d(x', x) between r and each instance (x, y) € D
- Choose K nearest training sets from r, that is, Dz € D
- Test instances are classified in accordance with most of the nearest neighbor classification categories

end For
END

Figure .1 The KNN Classifier Pseudo Code

To show the behavior of KNN results, BOW was applied as the extraction method. The dataset that we had used were taken spam email consisting of 4601 emails and Enron Spam Corpus (Enron1 to Enron 6) which contain 30041 emails that firstly divided into 50% training and 50% testing. This experiment used the fitness function as an evaluation measure. Figure 2 shows the real correlation for the 1000 iteration for KNN that has been constant result around 0.2 and the best result going to zero. The next sub-section will descript the other classifier called Naive Bayes.

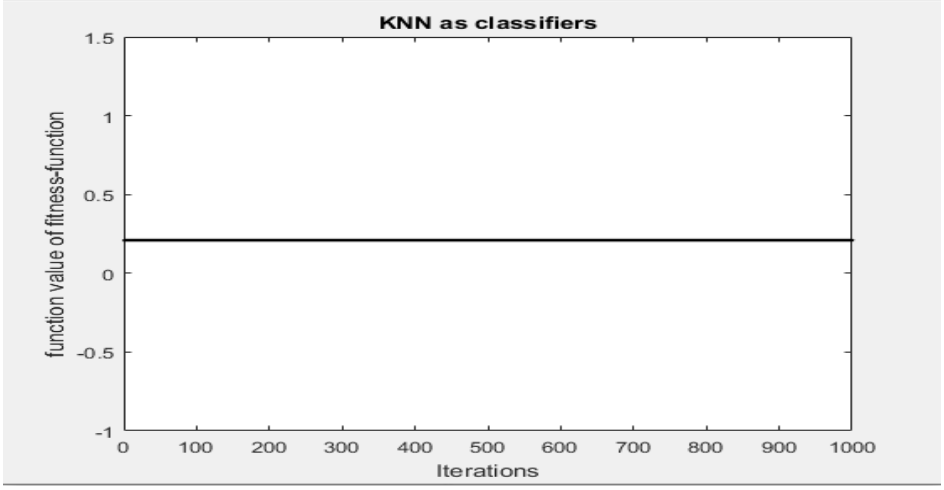

Figure. 2 Bow Extraction Using KNN for 1000 independent run

2.2 Naïve Bayesian

The NB is a probabilistic model which estimate the probabilities of categories in a given test document by exploiting the joint probabilities of terms and categories [21] The naive aspect of the classifier comes from the simple assumption of the conditional independence of all terms from each other given a category. Owing to this assumption, there is a chance of individually learning each term parameters; this makes the computation process simple and fast compared to non-naive Bayes classifiers.

Multinomial model and multivariate Bernoulli model have been discussed by [32] as the two common NB text classification models. Text classification in both models is performed by applying the Bayes' rule [21]:

$$P(cj\ |di) = \frac{P\ (c_j)\ P(d_i|c_j)}{d_i} \quad (2)$$

where di = the test document, and cj = a category.

Given the test document di, the posterior probability of each category cj, i.e. P (cj |di), is determined and the highest-ranking category in terms of probability is assigned to di. The calculation of P (cj |di) requires a pre-estimation of the P(cj) and P(di|cj) from the set of training documents. As P (di) is present to each category, it can be overlooked during the computation; hence, the category prior probability, P (cj), can be estimated thus:

$$\hat{}_P c_j = \frac{\sum_{i=1}^{N} y(\ d_i, c_j)}{N} \quad (3)$$

where N = number of training documents, and y (di, cj) is:

$$Y\ (di,\ cj) = \begin{cases} 1\ if\ d_i\ \in c_j \\ o\ otherwise \end{cases} \quad (4)$$

The prior probability of category cj can, therefore, be determined using the set of documents that belong to cj in the train set. The estimation of P(di|cj) parameters can be done in several ways using multinomial and multivariate Bernoulli models which will be described subsequently. Some of the previous works on the use of popular classifiers for spam classification are presented in Table1.The following Figure 3 below illustrates and shows completely the NB pseudo code.

- BEGIN: For all the available values,
- Follow the rules for each and every individual value as:
- Calculate and count the values of the classes appearing
- Obtain the class, which is occurring frequently
- Make the rule, which connects this particular class with instance values
- Find out the rate at which the error occurred for the rule
- Choose the rules with the smallest error rate END

Figure .3 The NB Classifier Illustrative Pseudo Code

In order to illustrate the behaviour of NB results, BOW was being applied as the original extraction method. were taken spam email consisting of 4601 emails and Enron Spam Corpus (Enron1 to Enron 6) which contain 30041 emails that firstly divided into 50% training and 50% testing. This This experiment used the fitness function as an evaluation measure. Figure 4 shows that the 1000 iteration for the NB concludes that has constant result around 0.6 and the best measure of fitness function is 0. The next sub-section will descript the last classifier will be evaluated in this chapter called Support Vector Machine, and will be compare with the result in two other classifiers mentioned before.

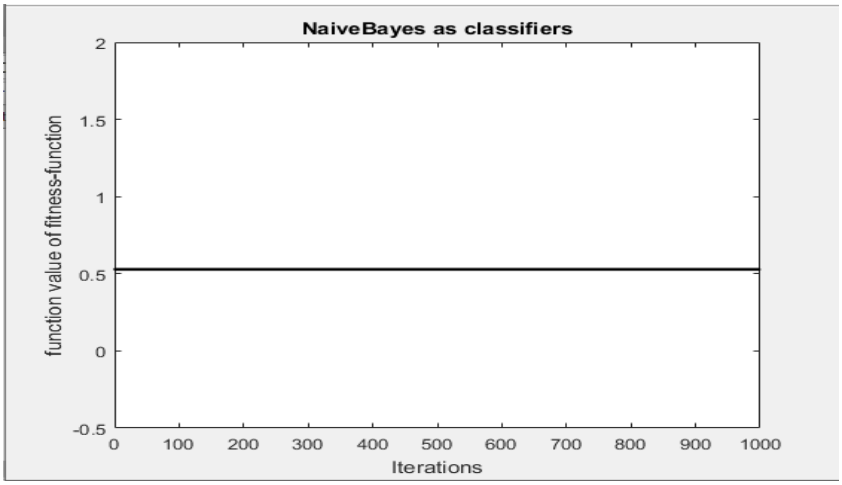

Figure .4 BOW Extraction Using the NB Classifier for 1000 independent run

2.3 Support Vector Machine

The SVM was introduced by [25] as a technique that depends on the Structural Risk Minimization principle [26] for providing solutions to two-class pattern recognition problems. However, a major challenge here is finding the decision surface which can maximally separate the training examples (positive and negative) of a category. In a linearly separable space, a decision surface is a hyperplane and the extent the decision surface can vary without impacting the classification process is represented by the dashed lines next to the solid one (where the distance between these parallel lines is the margin). Support vectors are examples which are closest to the decision surface.

As per [31] the decision surface for a linearly separable case is a hyperplane which can be written as:

$$w \bullet d + b = 0 \quad (5)$$

where d = the considered document, and w and b are to be learned from the training set.

The major challenge in the SVM is to find w and b which can meet the constraints [27]:

$$\text{Minimize } ||w||2 \quad (6)$$

$$\text{So that } \forall\, i : yi\,[w \bullet d + b] \geq 1 \quad (7)$$

Here, $i \in \{1, 2, ...,N\}$, where N represents the available documents in the training set; and yi = +1 if document di is a positive instance for the current category, and −1 if not. Quadratic programming techniques can be used to solve this optimization problem [27].

similarly, non-linear decision functions like radial basis function (RBF) with variance γ or polynomial of degree d can be leaned using SVM. The illustration of these kernel functions is as follows:

$$K_{polynomial}(d1, d2) = (d1 \bullet d2 + 1)d \quad (8)$$

$$K_{rbf}(d1, d2) = \exp(\gamma (d1 - d2)2 \quad (9)$$

In this study, the evaluated models are SVM with linear kernel, RBF kernel with different γ parameters, and polynomial kernel with different degrees. The SVM light system previously implemented by [33] was used in our experiment.

[28] performed spam classification using four feature selection on SVM and "bag of word" as extraction. The study reported the best type of SVM as the Gaussian Kernel which performed better than Polynomial Kernel and Linear Kernel SVM. A comparison of Feature Selection ConcaVe (FSV), Recursive Feature Elimination (RFE), and Fisher and Kernel-Penalized also showed Kernel-Penalized as the best.

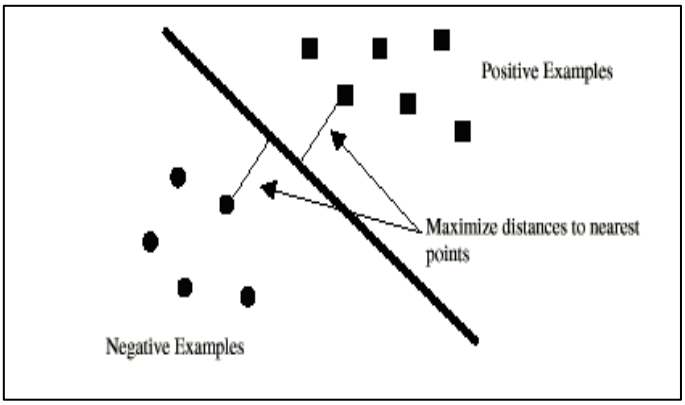

Figure .5 Data Values Classification

It is very clear from the above figure, which shows the 2D-dimensional case, in which the data where the data points are linearly separable is obtained.

In order to illustrate the behavior of SVM results, BOW rule had been applied as the extraction method. The datasets which were used, were taken from spam email consisting of 4601 emails and Enron Spam Corpus (Enron1 to Enron 6) which contain 30041 emails that firstly divided into 50% training and 50% testing This experiment can be used for the fitness function as an evaluation measure.

Figure 6 reveals the 50 iteration of the runs that was executed for SVM and that has constant result around 0.1 and the best measure of fitness function is 0, which can observe that the KNN better than NB but the SVM better than other two classifiers. That is motivate this paper to Evaluation of the performance of popular three classifiers on spam email classification to specify which one is the best.

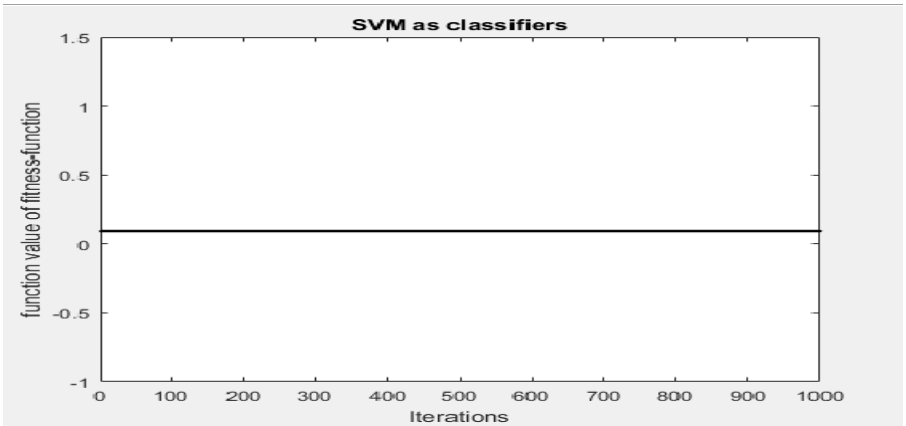

Figure .6 Bow Extraction Using SVM Classifier for 1000 independent run

Table1. show some of previous works in spam classifiers

Author	Year	Dataset	Extraction	With or without FS	Classifiers	Outperform classifiers
Parveen & Halse	2016	dataset of spam	BOW	Without	NB, SVM, and Tree-based J48	NB
Maldonado & Huillier	2013	Spam and Phishing	BOW	Recursive Feature Elimination, Feature Selection ConcaVe, Fisher and Kernel-Penalized	SVM + Polynomial Kernel and Linear Kernel	Kernel-Penalized + Gaussian Kernel
DeepaLakshmi. & Velmurugan	2016	SMS Spam Collection	BOW	Pearson correlation, Symmetric Uncertainty, Mutual Information, and Chi-square	NB	tree-based J48 and IB1
Diale et al	2016	Enron spam email	BOW	Information Gain and Chi-Square	AdaBoost, RF, and SVM	RF and SVM
Liu et al	2014	PU1, LingSpam, SpamAssian and Trec2007	BOW	with	SVM	NB
Mccord & Chuah	2011	1000 Twitter	BOW	without	RF, NB, SVM, and K-NN	RF

3 Results and Discussions

The quality of the classifiers was evaluated using three quality measures, namely f-measure, accuracy, and error rate [29]. Majorly, the external quality measure depends on the labeled test of the email corpora. It involves comparing the resulting classifiers and the labeled classes, then, measure the extent that emails from the same class are allocated/assigned to the same class. Accuracy, the commonly used measures in text mining, was used in this study as the external quality measure.

3.1 Accuracy

The absolute most interesting summary of classifier performance is the confusion matrix. This matrix is a table that summarizes the classifier's predictions against the known data categories. The confusion matrix is a table counting how often each combination of known outcomes (the truth) occurred in combination with each prediction type.

In classification problems, the evaluation measures are generally defined from a matrix with the numbers of examples correctly and incorrectly classified for each class (also known as the confusion matrix (CM)). Table 2 showed the CM for a binary classification task with only positive and negative classes.

Table 2. Confusion matrix

Predicted Class		
True Class	Positive	Negative
Positive	TP	FN
Negative	FP	TN

The accuracy rate (ACC) is the most common evaluation measure used in practice; it evaluates the effectiveness of a classifier based on the percentage of its correct predictions. The ACC equation is computed thus:

ACC = ((TP+TN)/ (TP+TN + FP+FN)) *100 (10)

3.2 F-measurement

This metric merge both the recall and precision ideas gained from information retrieval. With this measure, each class is taken as the results of emails and perceived as the ideal set of emails or spam. The calculation of the recall and precision for each email j and class i is done thus:

$$\text{Recall (i,j)} = \frac{n_{ij}}{n_i}. \quad (11)$$

$$\text{Precision (i,j)} = \frac{n_{ij}}{n_j}. \quad (12)$$

where n_{ij} is the number of available mails having the class label i in class j, n_i is the number of emails with the class label I, and n_j is the number of emails in class j. The calculation of the F-measure of email j and class i is done thus:

$$\text{F (i,j)} = \frac{2\text{Recall (i,j)Precision (i,j)}}{\text{Recall (i,j)+Precision (i,j)}}. \quad (13)$$

The calculation of the cumulative F-measure measure is done by considering the weighted average value of the component F-measures as follows:

$$\text{F}=\sum_i \frac{n_i}{N} maxF(i,j). \quad (14)$$

Therefore, the F-measure values are observed to be in the range of (0,1); larger +9values = better classifier quality.

3.3 Data Sets Used in The Experiments

A comprehensive analysis of the deployed datasets in several email applications and classification is presented in this section. Email classifiers are mainly used in the classification of spam emails, phishing emails, spam and phishing emails, and multi-folder emails categorization. Hence, this study employed public datasets to further investigate these areas. Table 3.2 presents a detailed analysis of various datasets used in various applications.

As per [30] most of the popular dataset used in spam email classification are Spam-Base datasets (eight studies), Spam Assassin (five studies), and Enron spam email corpus (five studies). Most of the studies use real email subsets (sourced from the existing spam datasets) for real email messages. Among the 9 studies reported on phishing email classification applications, phishing corpus with Spam Assassin dataset was used in 8 of the studies. Moreover, the emails provided information about the types of materials sorted as they contain different types of phishing approaches. [30] reported the use of PhishingCorpus for spam and phishing email classification, and a combination of LingSpam, SpamBase, and SpamAssassin datasets for spam detection.

Among 20 studies reported on multi-folder email categorization, 6 of the studies used Enron dataset; this is a widely used dataset in multi-folder categorization due to its wide availability in email classification. It

contained 252,757 pre-processed emails of 151 employees in 3,893 folders [30]. However, the Enron spam corpus differed from the Enron email datasets as the former is a successor of both Ling-Spam and Enron email datasets [30] All the publicly available datasets used in different email classification areas (together with their available links) are presented in Table 3 that include DS1 which called spam base we take it from UCI and the total of emails is 4601 and the number of spam= 1813 and ham=2788 and DS2 which called Enron Spam Corpus and it contains 6 datasets and the total of emails are30041 and the number of spam= 13496 and ham=16545.

Table 3. Summary of spam datasets

Document set	Source	Number of emails
DS1	Spam Base http://archive.ics.uci.edu/ml/data sets/Spambase	Total 4601 emails (spam = 1813 and ham = 2788)
DS2	Enron Spam Corpus http://www.aueb.gr/users/ion/dat a/enron-spam/	Total 30041 emails (spam = 13496 and ham = 16545)

3.4 Results

The section compared the classification accuracies (in percentage) of using K-NN, NB, and SVM classifiers in email classification; the number of features before using the feature selection method was also presented. Furthermore, the results from experiments without any form of attribute reduction were presented as well. From the results based on f-measure and classification accuracy, SVM performed a better classification (using seven datasets) compared to KNN and NB. However, the SVM cannot be claimed to consistently perform better than KNN and NB. Regarding the number of features, they all had a similar number of features (note that using all the feature does not guarantee absolute accuracy likely because of the presence of irrelevant/redundant attributes in the datasets.

Table 4. The classification accuracy using K-NN, SVM, and NB

Dataset	K-NN	SVM	NB
Enron1	85.200000	83.8000	46.400000
Enron2	73.400000	87.0000	46.000000
Enron3	63.833333	81.8333	49.000000
Enron4	70.666667	78.6667	50.666667
Enron5	80.7000	87.3000	53.100000
Enron6	66.0000	75.1000	48.500000
Spam Base	74.096045	79.039	46.977401

Table 5. The number of features using K-NN, SVM, and NB

Dataset	K-NN	SVM	NB
Enron1	16383	16383	16383
Enron2	11514	11514	11514
Enron3	16382	16382	16382
Enron4	15456	15456	15456
Enron5	14696	14696	14696
Enron6	16380	16380	16380
Spam Base	57	57	57

Table 6. The results of f-measure using K-NN, SVM, and NB

Dataset	K-NN	SVM	NB
Enron1	84.6109	84.8668	46.3973
Enron2	66.8329	89.3628	48.6192
Enron3	71.1753	81.9624	48.5405
Enron4	56.0000	78.6879	53.8986
Enron5	77.6402	86.1295	53.8432
Enron6	72.0713	74.8350	50.2027
Spam Base	72.9517	79.7439	48.6453

Table 4,6 show a summary of the performance results for the three classifiers namely KNN, NB, and SVM were tested in MATLAB using seven email spam datasets. The results presented were based on the accuracy and f-measurement achieved using the BOW extraction method. From the results, the SVM performed a better classification in all datasets except (Enron1) the KNN gave the best result and (NB achieved the least classification performance). Table 5 shows the number of features in all datasets which is the same in all classifiers before reduction or use of feature selection methods.

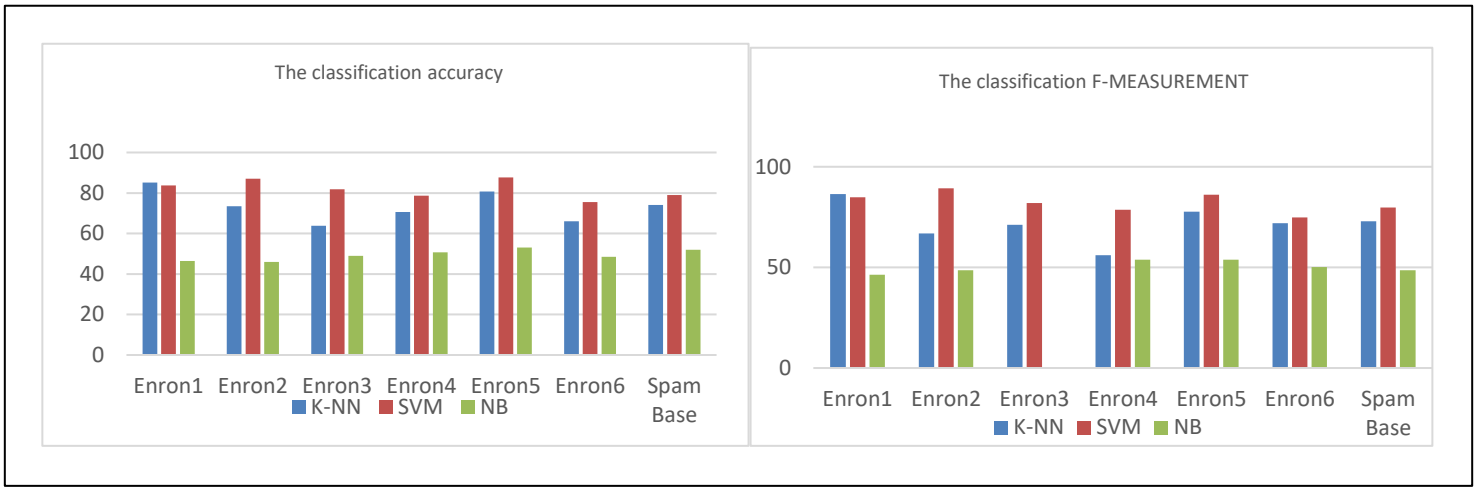

Figure .7 Graphical representation of results

Based on the results of Table 4, which illustrated the accuracy, and Table 6 which showed the f-measurement. Figure 7 indicates that the SVM classifier achieved the best results in all datasets which gave the best accuracy of 87.3%. except for Enron1, which gave the best accuracy with KNN.

4 Conclusions

This paper reported an experiment on the determination of the classification accuracy of three classifiers in email classification using a MATLAB platform. The aim is to establish a better classifier that can specifically assign emails into the right class (spam or non-spam). The three classifiers (K-NN, SVM, and NB) were compared based on different performance measures. Classification problem involves the rightful identification of an object and its subsequent placement in the right class (for instance, classifying a given email as spam or non-spam). From the simulation results, the SVM achieved the best classification accuracy when using spam email datasets with 7 datasets, followed by K-NN. The least performance was observed with NB as it showed lower accuracy and f-measure compared to SVM and K-NN. This observation showed that SVM is a better classifier for spam email application especially when classification accuracy is paramount. Future studies will consider an extension of the simulation studies performed in the MATLAB platform to reduce the number of features. This can be achieved by deploying an optimization technique during the feature selection process.

Reference

[1] Zdziarski, J.A., 2005. Ending spam: Bayesian content filtering and the art of statistical language classification. No Starch Press.

[2] Al-Gasawneh, J., & Al-Adamat, A. (2020). The mediating role of e-word of mouth on the relationship between content marketing and green purchase intention. Management Science Letters, 10(8), 1701-1708.

[3] El Kourdi, M., Bensaid, A. and Rachidi, T.E., 2004, August. Automatic Arabic document categorization based on the Naïve Bayes algorithm. In proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages (pp. 51-58). Association for Computational Linguistics.

[4] Al-Harbi, S., Almuhareb, A., Al-Thubaity, A., Khorsheed, M.S. and Al-Rajeh, A., 2008. Automatic Arabic text classification.

[5] Mafarja, Majdi M., and Seyedali Mirjalili. "Hybrid Whale Optimization Algorithm with simulated annealing for feature selection." Neurocomputing 260 (2017): 302-312.

[6] Eyheramendy, S., Lewis, D.D. and Madigan, D., 2003. On the naive bayes model for text categorization.

[7] Galavotti, L., Sebastiani, F. and Simi, M., 2000, September. Experiments on the use of feature selection and negative evidence in automated text categorization. In International Conference on Theory and Practice of Digital Libraries (pp. 59-68). Springer, Berlin, Heidelberg.

[8] Duwiri, M.U.C., 2007. KERAGAMAN JENIS DAN PENYEBARAN KUPU–KUPU SUPERFAMILI PAPILIONOIDEA ORDO LEPIDOPTERA DI KAMPUNG MOKWAM DISTRIK MINYAMBOU KABUPATEN MANOKWARI (Doctoral dissertation, Universitas Negeri Papua).

[9] Forman, G., 2003. An extensive empirical study of feature selection metrics for text classification. Journal of machine learning research, 3(Mar), pp.1289-1305.

[10] Fragoudis, D., Meretakis, D. and Likothanassis, S., 2005. Best terms: an efficient feature-selection algorithm for text categorization. Knowledge and Information Systems, 8(1), pp.16-33.

[11] Gupta, A.K. and Nagar, D.K., 2018. Matrix variate distributions. Chapman and Hall/CRC.

[12] Sarkar, J.L., Panigrahi, C.R., Pati, B., Trivedi, R. and Debbarma, S., 2018. E2G: A game theory-based energy efficient transmission policy for mobile cloud computing. In Progress in Advanced Computing and Intelligent Engineering (pp. 677-684). Springer, Singapore.

[13] Ozgur, A., 2004. Supervised and unsupervised machine learning techniques for text document categorization. Unpublished Master's Thesis, İstanbul: Boğaziçi University.

[14] Wang, Y., Bryant, S.H., Cheng, T., Wang, J., Gindulyte, A., Shoemaker, B.A., Thiessen, P.A., He, S. and Zhang, J., 2016. Pubchem bioassay: 2017 update. Nucleic acids research, 45(D1), pp.D955-D963.

[15] Jain, M., Koren, S., Miga, K.H., Quick, J., Rand, A.C., Sasani, T.A., Tyson, J.R., Beggs, A.D., Dilthey, A.T., Fiddes, I.T. and Malla, S., 2018. Nanopore sequencing and assembly of a human genome with ultra-long reads. Nature biotechnology, 36(4), p.338.

[16] Saha, S.K., Ghasempour, Y., Haider, M.K., Siddiqui, T., De Melo, P., Somanchi, N., Zakrajsek, L., Singh, A., Shyamsunder, R., Torres, O. and Uvaydov, D., 2019. X60: A programmable testbed for wideband 60 ghz wlans with phased arrays. Computer Communications, 133, pp.77-88.

[17] Parveen, P., 2016. Prof. Gambhir Halse, "Spam mail detection using classification". International Journal of Advanced Research in Computer and Communication Engineering, 5(6).

[18] DeepaLakshmi, S. and Velmurugan, T., 2016. Empirical study of feature selection methods for high dimensional data. Indian Journal of Science and Technology, 9, p.39.

[19] Pletcher, R.H., Tannehill, J.C. and Anderson, D., 2012. Computational fluid mechanics and heat transfer. CRC press.

[20] Wijayawardene, N.N., Crous, P.W., Kirk, P.M., Hawksworth, D.L., Boonmee, S., Braun, U., Dai, D.Q., D'souza, M.J., Diederich, P., Dissanayake, A. and Doilom, M., 2014. Naming and outline of Dothideomycetes–2014 including proposals for the protection or suppression of generic names. Fungal Diversity, 69(1), pp.1-55.

[21] Mitchell, R.J., 1997. Effects of pollen quantity on progeny vigor: evidence from the desert mustard Lesquerella fendleri. Evolution, 51(5), pp.1679-1684.

[22] Yang, Y. and Liu, X., 1999, August. A re-examination of text categorization methods. In Sigir (Vol. 99, No. 8, p. 99).

[23Re, R., Pellegrini, N., Proteggente, A., Pannala, A., Yang, M. and Rice-Evans, C., 1999. Antioxidant activity applying an improved ABTS radical cation decolorization assay. Free radical biology and medicine, 26(9-10), pp.1231-1237.

[24] Mccord, M. and Chuah, M., 2011, September. Spam detection on twitter using traditional classifiers. In international conference on Autonomic and trusted computing (pp. 175-186). Springer, Berlin, Heidelberg.

[25] Drucker, H., Burges, C.J., Kaufman, L., Smola, A.J. and Vapnik, V., 1997. Support vector regression machines. In Advances in neural information processing systems (pp. 155-161). [26] Burges, C.J., 1998. A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery, 2(2), pp.121-167.

[27] Joachims, T., 1998, April. Text categorization with support vector machines: Learning with many relevant features. In European conference on machine learning (pp. 137-142). Springer, Berlin, Heidelberg.

[28Maldonado, S. and L'Huillier, G., 2013. SVM-based feature selection and classification for email filtering. In Pattern recognition-applications and methods (pp. 135-148). Springer, Berlin, Heidelberg.

[29] Karypis, M.S.G., Kumar, V. and Steinbach, M., 2000, August. A comparison of document clustering techniques. In TextMining Workshop at KDD2000 (May 2000).

[30] Mujtaba, G. and Lee, K., 2017. Treatment of real wastewater using co-culture of immobilized Chlorella vulgaris and suspended activated sludge. Water research, 120, pp.174-184.

[31] Yang, Y. and Liu, X., 1999, August. A re-examination of text categorization methods. In Sigir (Vol. 99, No. 8, p. 99).

[32] McCallum, A. and Nigam, K., 1998, July. A comparison of event models for naive bayes text classification. In AAAI-98 workshop on learning for text categorization (Vol. 752, No. 1, pp. 41-48).

[33] Joachims, T., 1999. Svmlight: Support vector machine. SVM-Light Support Vector Machine http://svmlight. joachims. org/, University of Dortmund, 19(4).

[34] Al-Gasawneh, J. A., & Al-Adamat, A. M. (2020). THE RELATIONSHIP BETWEEN PERCEIVED DESTINATION IMAGE, SOCIAL MEDIA INTERACTION AND TRAVEL INTENTIONS RELATING TO NEOM CITY. Academy of Strategic Management Journal, 19(2).

[35] Dagher, Issam, and Rima Antoun. "Ham-spam filtering using different PCA scenarios." In 2016 IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES), pp. 542-545. IEEE, 2016.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)