

Comparing Supervised Learning Classifiers to Detect Advanced Fee Fraud Activities on Internet

Abiodun Modupe¹, Oludayo O. Olugbara², and Sunday O. Ojo³

¹ Department of Information Technology, Tshwane University of Technology,
Pretoria, South Africa

² Department of Information Technology, Durban University of Technology,
Durban, South Africa

³ Faculty of Information Technology, Tshwane University of Technology,
Pretoria, South Africa

{modupea,ojoso}@tut.ac.za, oludayoo@dut.ac.za

Abstract. Due to its inherent vulnerability, internet is frequently abused for various criminal activities such as Advanced Fee Fraud (AFF). At present, it is difficult to accurately detect activities of AFF defrauders on internet. For this purpose, we compare classification accuracies of Binary Logistic Regression (BLR), Back-propagation Neural Network (BNN), Naive Bayesian Classifier (NBC) and Support Vector Machine (SVM) learning methods. The word clustering method (globalCM) is used to create clusters of words present in the training dataset. A Vector Space Model (VSM) is calculated from words in each e-mail in the training set. The WEKA data mining framework is selected as a tool to build supervised learning classifiers from the set of VSMs using the learning methods. Experiments are performed using stratified 10-fold cross-validation method to estimate classification accuracies of the classifiers. Results generally show that SVM utilizing a polynomial kernel gives the best classification accuracy. This study makes a positive contribution to the problem of detecting unwanted e-mails. The comparison of different learning methods is also valuable for a decision maker to consider tradeoffs in method accuracy versus complexity.

Keywords: Advanced Fee Fraud, Word Clustering, Supervised Learning, Cluster Features.

1 Introduction

The objectives of this study are to discover a set of features and an effective learning classifier to accurately detect AFF activities on internet. Despite numerous benefits that internet technology offers to mankind, the information system is open to sabotage. Many crimes, including AFF, identity theft, telemarketing, insurance fraud, cyber squatting, cyber stalking, online gambling, lottery fraud and investment scams are perpetrated on internet. The occurrence of these cybercrimes has negative consequences on security of individuals and compromised security of internet technology.

AFF is a social engineering scheme wherein the defrauder requests a cash advance to facilitate a much greater payoff. This is a variant of the Spanish prisoner scam, which is now known as Nigerian-419. This scam is notoriously based in African countries mostly in Nigeria, Ghana, South Africa and Cameroun [1]. The tactics of defrauders reside in the bulk of email messages to find promising gullible individuals who can be easily tempted by quick financial reward. For instance, AFF emails describe the need under different pretexts to move a huge sum of money across a country. AFF defrauders feel untouchable and secure that they routinely impersonate government authorities and multinational corporations to defraud individuals. AFF activities are malevolence and depressing trades that have constituted a nuisance to national security and prosperity of many individuals. Indeed, sophisticated AFF activities are conducted through the distribution of physical mail, fax and more recently, email messages. The information content is subjected to remote association, inheritance, over-budgeted contract payments, job offers, joint ventures, awards, lotteries and upfront fees for loans.

The US-based Internet Crime Complaint Center (IC3), which is a conglomeration of the National White Collar Crime Center (NW3C), Bureau of Justice Assistance (BJA) and Federal Bureau of Investigation (FBI) received more than three hundred thousand complaints in 2009 [2, 3]. Approximately, 43.56% of these complaints revolved around financial frauds. The total monetary lost for victims is in excess of \$559 million. This moves up from \$295 million lost reported in 2008. About 9.8% of complaints are reported to be cases of AFF and victims said they were contacted through internet. AFF scam recorded high lost next to FBI scam and non-delivery merchandise [2, 3]. This implies that cost of cybercrimes is constantly increasing and internet facilities are mainly used to facilitate these erroneous crimes. The internet seems to be a safe place for carrying out fraudulent and illegal business. This is because the society of today is heavily dependent on internet technology for different kinds of activities. Criminals are also exploiting numerous opportunities provided by internet technology to perpetrate their malevolence tendencies.

In the light of increasing cybercrimes, a Computer Forensic Competency (CFC) has been established to assist law enforcement agencies in cybercrime investigations [4, 5]. The onus of CFC is to investigate digital scenes by finding relevant facts in form of electronic evidence. These facts are to be presented in a coherent way to prosecute defrauders in law courts. Being able to accurately detect AFF activities on internet can be beneficial in many ways. For example, it would be possible to design intelligent systems to proactively detect and filter out malicious emails to reduce the modus operandi of defrauders. In addition, detecting AFF activities on internet can help to increase confidence and trust levels of individuals to engage in diverse electronic business transactions.

The remainder of this paper is succinctly summarized as follows. In Section 2, we describe related study. In Section 3, we discuss supervised learning classifiers that are compared to detect AFF activities on internet. In Section 4, we discuss methodology of this study. In Section 5, we discuss results of experiments performed. In Section 6, we give a concluding remark.

2 Related Study

The majority of existing techniques for spam message identification differ from one another for several reasons. Spam messages are of diverse forms including AFF, cyber-phishing, drug trafficking, cyber-bullying, sexual harassment and child pornography. As a result, no unified algorithms can accurately detect all of these spam types simultaneously for the following reasons. First, the primary tactic of defrauders is to hide their intent in order to influence an individual of a higher payoff. Second, spam messages are well engineered to read regular emails and successfully pass filters, antivirus, firewalls and scammers tests. Third, diverse messages can originate from the same individual and messages are not equivalent in contents. Fourth, spam messages are not necessarily sent through the same physical path or using the same algorithms.

Chandrasekaran et al [6] develop a technique that detects phishing emails from legitimate emails based on structural attributes such as linguistic properties, vocabulary richness and email subjects. They model 25 features ranked by information gain and tested the model with 200 emails (100 phishing and 100 legitimate). They use SVM to classify phishing emails based on these features. Results of the study show 95% classification accuracy rate with a low false positive prediction. The work reported in [7] uses BLR, SVM, Random Forest (RF), Bayesian Adaptive Regression Tree (BART) and Classification and Regression Tree (CART) learning classifiers to classify phishing emails. They use 43 features to train and test these classifiers. Results of the study show that it is difficult to evaluate prediction accuracy using one evaluation metric when BLR, SVM, RF, BART and CART are used as classifiers.

Fette et al [8] implemented RF learning classifier in a PILFER algorithm to detect phishing email from a corpus of 860 phishing emails and 6950 legitimate emails. Result of the study shows that 96% of phishing emails was correctly predicted with a false positive rate of 0.1%. Airoidi and Malin [9] classify emails into scam, spam and ham by comparing Poisson filter and Spam-Assassin to detect fraudulent hidden scam emails based on words extraction. Hadjidj et al [10] developed a technique to assist forensic investigators to collect clues and evidence in an investigation. Stylometric features such as lexical, syntactic and idiosyncratic were used to identify authors of malicious emails. They used Decision Tree (DT) and SVM learning classifiers coupled with integration of social network algorithm.

In our previous study [11], a model of identifying activities of AFF defrauders was introduced. A training dataset set of 1100 emails of which 680 emails belong to AFF class was used to train BLR and SVM to predict AFF emails. In this current study, we increase dataset size to 2000, add two more classifiers and use more evaluation metrics such as area under receiver operating characteristic curve, cross-validation and cost sensitive analysis with weighted accuracy. This approach is unique because it uses globalCM algorithm [12] to discover a set of cluster features that characterizes AFF activities on internet. Preprocessing of emails using a combination of lemmatization and globalCM algorithms to create clusters of semantically related word is a valuable insight that can be applied to multiple problems in text classification.

3 Supervised Learning Classifiers

A Supervised Learning Classifier (SLC) solves a machine learning task of inferring a function from available example data. The purpose is to predict the desired supervisory signal or output for a valid input vector. A learning or classification task is a famous data mining problem that can be defined as a process of assigning a class label to a data instance, given a set of previously classified data instances. Two basic processes of a SLC are training and testing. During the training phase, parameters associated with the learning model are updated based on inputs received from the environment. During the testing phase, a new input vector whose class is probably unknown is presented to the classifier to predict the appropriate class the input vector belongs. This study experimentally compares BLR, BNN, SVM and NBC learning methods to detect AFF activities on internet.

3.1 Binary Logistic Regression

Binary Logistic Regression (BLR) assumes that a logistic or sigmoid relationship exists between probability of group membership and one or more features [13]. The BLR model is used to relate probabilities of group membership to a linear function of data features. The probability values $p_1(F)$ and $p_2(F)$ that a data instance $F = (F_1 = f_1, F_2 = f_2, \dots, F_n = f_n)$ of n features belongs to sample groups 1 and 2 respectively is given in terms of logit transform of odds ratio as:

$$\ln\left(\frac{p_1(F)}{p_2(F)}\right) = a_0 + a_1f_1 + a_2f_2 + \dots + a_nf_n \quad (1)$$

where f_n is the value of n^{th} feature of the data instance, a_n is the coefficient of f_n and (a_0, a_1, \dots, a_n) are parameters of the logistic model. These parameters are usually estimated during training phase using maximum likelihood method.

Equation (1) can be rewritten to directly express the probability that the data instance belongs to group 1 as follows

$$p_1(F) = \frac{1}{1 + \exp(-a_0 - a_1f_1 - \dots - a_nf_n)} \quad (2)$$

The estimated regression coefficients (a_1, \dots, a_n) and constant a_0 are used to define a logistic model. The constructed model is used during testing to classify a new data instance into one of the two groups. The classification rule is usually based on a probability threshold of 0.5. If $p_1(F) \geq 0.5$, the data instance is classified into group 1 otherwise it is classified into group 2.

3.2 Backpropagation Neural Network

Backpropagation Neural Network (BNN) is a supervised learning classifier that generalizes delta rule and it learns through backward propagation mechanism. The network model provides great flexibility in linear speed so that each element can compare its input value against stored examples [14]. A decision function is usually chosen during network training from a family of functions that are represented by the network architecture. This family of functions is defined by complexity of the network according to the number of neurons in input and hidden layers of the network [15]. The decision function is determined by choosing suitable sets of weights for the network. The training process involves calculation of input and output values, activation and target functions, backward propagation of the associated error, adjustment of weight and biases [16]. The standard BNN model with a single output neuron can be represented as [15]:

$$y = \tilde{g} \left(\sum_{j=1}^m w_{1j}^2 \times g \left(\sum_{i=1}^d w_{ji}^1 \times x_i + w_{jo}^1 \right) + w_{11}^2 \right) \quad (3)$$

The input function \tilde{g} is usually represented by a linear function. The output function on hidden and output layer units is assumed to be sigmoid or tan-sigmoid. A typical sigmoid transfer function is the following bipolar activation.

$$f(x) = \frac{1 - \exp(-x)}{1 + \exp(-x)} \quad (4)$$

Sets of optimal weights are required to minimize the error function of the network, which represents deviation of predicted values y_k from observed $y(x_k)$ values.

The mean absolute error in output layer can be calculated as:

$$E_m = \frac{1}{2} \sum_{i=1}^n \sqrt{(y(x_k) - y_k)^2} \quad (5)$$

Where n is the number of training instances. The training of a network is typically performed on variations of gradient descent based algorithm to minimize error function [17]. In order to avoid over-fitting problem, cross-validation method is used to find an earlier point of training [18].

3.3 Support Vector Machine

Support Vector Machine (SVM) [19] is a supervised learning classifier, which is particularly suited for solving binary classification problems. SVM method is widely used because of its ability to handle high-dimensional data through the use of kernels. Given a training dataset $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, where each element of the dataset is represent by n-dimensional vector $x_i = (x_1^i, x_2^i, \dots, x_n^i)$

and $y_i \in \{-1,1\}$. The classifier proceeds to find a separating hyperplane $\{x|w^T x + b = 0\}$ that generates the largest margin between data points in positive and negative classes. This is achieved by solving optimal hyperplane problem, which is the solution of the following minimization problem [20]:

Minimize

$$\frac{1}{2} \|w\|^2 + c \sum_{i=1}^n \xi_i \tag{6}$$

Subject to the constraints:

$$\begin{cases} y_i (w^T x_i + b) \geq 1 - \xi_i, \\ \xi_i \geq 0, i = 1, \dots, n \end{cases} \tag{7}$$

where ξ_i represents a slack-variable that allows misclassification to occur and C is a trade-off parameter for generalization performance.

The basic assumption of SVM is that training dataset is linearly separable. This is not generally the case in reality as training dataset can contain data points that are linearly inseparable. The solution therefore, is to transform non-linear dataset into the one that is linear using kernel functions $K(x_i, x_j)$ such as linear network, polynomial, radial-basis and two-layer perceptions. The ideal of kernels is to enable operations to be performed in the input space instead of the potentially high dimensional feature space [21]. The SVM classification task is a quadratic programming optimization problem that can be solved through kernel based dual formulation to maximize the following performance function.

Maximize

$$J(\alpha) = \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m y_i y_j \alpha_i \alpha_j K(x_i, x_j) \tag{8}$$

Subject to the constraints:

$$\sum_{i=1}^m y_i \alpha_i = 0, (0 \leq \alpha_i \leq C) \tag{9}$$

where $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_m\}$ are Lagrangian variables to be optimized and m is the number of training instances. The decision function of the classifier is given by:

$$f(x) = sign\left(\sum_{i=1}^n y_i \alpha_i K(x_i, x_i) + b\right) \tag{10}$$

where $n \leq m$ is number of support vectors ($\alpha_i > 0$) and b is the bias that satisfied the Karush-Kulu-Tucker optimality constraints of the dual formulation problem.

3.4 Naïve Bayesian Classifier

Naive Bayesian Classifier (NBC) is a probabilistic classifier used extensively to solve text classification problems [21]. During training phase, NBC learns class posterior probabilities $P(F_1 = f_1, F_2 = f_2, \dots, F_n = f_n | C = c_j)$ of each data feature F_i given the class label c_j , where $j = 1, 2$ is the number of classes and $i = 1, 2, \dots, n$ is the number of features. A new data instance of an n -dimensional vector of features values $F = (F_1 = f_1, F_2 = f_2, \dots, F_n = f_n)$ is classified into one of the two classes by the classifier. Bayesian rule is applied to compute posterior probability of each class c_j as follows.

$$\frac{P(C = c_j | F_1 = f_1, F_2 = f_2, \dots, F_n = f_n) = P(C = c_j) P(F_1 = f_1, F_2 = f_2, \dots, F_n = f_n | C = c_j)}{P(F_1 = f_1, F_2 = f_2, \dots, F_n = f_n)} \quad (11)$$

The Bayesian conditional independent assumption of features allows the simplification of class posterior probabilities to be expressed as product of each posterior probability of a feature for the given class. The class posterior probabilities are otherwise impossible to be estimated in reality because of data sparseness problems that can result from large samples. Thus, it follows that:

$$\frac{P(C = c_j | F_1 = f_1, F_2 = f_2, \dots, F_n = f_n) = P(C = c_j) \prod_{i=1}^n P(F_i = f_i | C = c_j)}{P(F_1 = f_1, F_2 = f_2, \dots, F_n = f_n)} \quad (12)$$

Class prior probabilities $P(F_1 = f_1, F_2 = f_2, \dots, F_n = f_n)$, $P(C = c_j)$ and posterior probability $P(F_i = f_i | C = c_j)$ are easy to estimate from available training dataset as frequency ratios. The law of total probability allows the class posterior probability that an email represented as a feature vector belongs to a class c_j to be expressed as:

$$\begin{aligned}
P(C = c_j | F_1 = f_1, F_2 = f_2, \dots, F_n = f_n) = \\
\frac{P(C = c_j) \prod_{i=1}^n P(F_i = f_i | C = c_j)}{\sum_{c_k \in C} P(C = c_k) \prod_{i=1}^n P(F_i = f_i | C = c_k)}
\end{aligned} \tag{13}$$

The decision rule of NBC assigns a new data instance to the class c_k ($k = 1, 2$) with the highest class posterior probability. In order to maximize the class posterior probability, the prior probability $P(F_1 = f_1, F_2 = f_2, \dots, F_n = f_n)$ will not be calculated because it serves as a normalizing factor and is constant for both classes. Thus, it follows that:

$$\begin{aligned}
P(C = c_k | F_1 = f_1, F_2 = f_2, \dots, F_n = f_n) = \\
\arg \max_{c_j} \left\{ P(C = c_j) \prod_{i=1}^n P(F_i = f_i | C = c_j) \right\}
\end{aligned} \tag{14}$$

4 Methodology

The methodology of this study consists of the sequence of actions that must be completed to realize the objectives of the study. In order to meet the objectives, two different essential tasks are to be performed. The tasks are to discover a set of features and an effective learning classifier that can assist to accurately detect AFF activities on internet. Email preprocessing and classification are two important steps of our methodology. Emails considered in this study are assumed to be written in English language.

The email processing procedure strips all attachments to facilitate extraction of contents of header and body from incoming email and its attachment if any. Html tags, video clip and image elements are extracted from email body. The algorithm then performs tokenization to extract words in email body. The process of lemmatization or stemming is performed to group morphological variants of the same words into their canonical form or stem. Porter stemming algorithm [22] is used to remove commoner morphological and inflexional suffixes. For example, stemming algorithm reduces the word forms banks, banking, banker and bankers to their stem bank to improve classification accuracy and AFF vocabularies. A more sophisticated procedure such as concept signatures [23] can also be used, but Porter algorithm is widely used for text stemming. In addition, the preprocessing algorithm removes stop-words and noisy words that often occur in text messages. Precisely 582 English stop-words are removed in this study. The globalCM algorithm [12] is finally used to compute cluster features by partitioning set of distinct words into clusters of semantically related words. This results in a saving of storage space and improves computational time efficiency as cluster features give compact representation of sets of semantically related words.

The email classification procedure implements both training and testing functions to make a Boolean decision on labelled email instances, wherein labels are AFF and not-AFF. We develop training dataset by performing a random selection of 980 AFF emails collected between April 2000 and June 2005 published on polifos¹ and svbizlaw² websites. This set of emails covers many of the recent trends in AFF business. For legitimate portion of the dataset, we use 1020 emails selected from Enron corpus [24]. Based on the cluster features discovered, a Vector Space Model (VSM) [25] representation is then calculated from the words in each e-mail in the training set. Precisely 42 cluster features were discovered to characterize AFF activities on internet. As a result, our dataset contains 2000 emails represented by VSM with 42 cluster features as dataset fields. The dataset is given as an input to the classifiers to build classification models that are validated with a set of testing examples.

The WEKA [26] data mining tool is used to build classifiers from our dataset using BLR, BNN, SVM and NBC learning methods. The Java based implementation of our experimentation system has a function to convert a dataset into WEKA compatible Attribute-Relation File Format (ARFF). This provides us with a simple means to interface Java based system with WEKA tool. The integration of an open source specialized machine learning program into our system gives us flavour of reliability and robustness. We used stratified 10-fold cross-validation method to obtain an estimate of the generalized error of all classifiers. K-fold cross-validation method is generally used to estimate performance of a model [27]. The cross-validation method works like this, the dataset is divided into k folds, in our experiment $k = 10$. A single fold is chosen as testing data and the remaining $k - 1$ folds are used for training. The process is repeated k times so that each k fold is used exactly once for testing.

5 Results and Discussion

The WEKA tool is used to perform a test to find minimum average error rate for BNN. Different number of units are used in hidden layer with wildcard values 'a' = (attributes + classes)/2, 'i'=attributes, 'o'=classes only and 't'=(attributes + classes), for 2, 22, 42 and 44 units. In addition, we use different weight decays of typical values of 0.1, 0.2, 0.25, 0.3, 0.4 and 0.5 respectively on interconnections. Results of the experiment show that BNN with size 22 and weight decay of 0.3 at epochs 30000 provides the lowest error rate of 0.009. We then use BNN model to establish comparison with other learning methods. Moreover, we obtain minimum average error rate for SVM using all kernel functions in WEKA. Polynomial kernels of degrees 2 and 3 with widths of 0.001, 0.005, 0.01, 0.05, 0.1, 0.5, 1 and 2 give the best performance. This is when compared to other learning methods with maximum optimal that generalized parameter factor of ten in order of $10^{-7} - 10^3$ for each

¹ <http://potifos.com/fraud>

² <http://www.svbizlaw.com/nigerian.419.letters.htm>

kernel. For BLR, we use 10 values of k ranging from 1 to 10 and Euclidean distance weighted as a gain ratio to train the classifier.

Experiments are performed for both regularized and un-regularized classifiers by varying the regularization parameter by factors of 10. The essence of regularization is to reduce over-fitting problems that are often associated with learning methods. The cost sensitive measures of weighted accuracy (WA_{cc}) and weighted error (WA_{err}) were used to compare classification accuracies of classifiers. Table 1 shows this result with the corresponding standard deviation (STDEV) when $\lambda = 1$ (legitimate and AFF emails are equally weighted) and $\lambda = 9$ (false positives are penalized nine times more than false negatives). The error rate of each classifier is calculated based on average error rate of all 10-fold samples with equivalent STDEV. This result shows that accuracy exceeded 90% for all classifiers when false positive had an equal penalty to false negative, but dropped to below 75% when false positive has a penalty of nine times that of false negative. SVM has the lowest WA of 0.0329 when $\lambda = 1$ and BNN has the lowest WA_{err} of 0.0273 when $\lambda = 9$.

Table 1. WA_{cc} and WA_{err} when $\lambda = 1$ and $\lambda = 9$ for all classifiers\

Classifier	$\lambda = 1$				$\lambda = 9$			
	WA_{cc}		WA_{err}		WA_{cc}		WA_{err}	
	AVG	STDEV	AVG	STDEV	AVG	STDEV	AVG	STDEV
BNN	0.9635	0.0025	0.0365	0.0025	0.7271	0.0021	0.273	0.0021
SVM	0.9671	0.0034	0.0329	0.0034	0.7247	0.0010	0.275	0.0010
BLR	0.9635	0.0013	0.0367	0.0013	0.7249	0.0018	0.275	0.0017
NBC	0.9301	0.0008	0.0698	0.0008	0.7197	0.0005	0.280	0.0005

In email classification problem, False Positive (FP) is the set of legitimate emails that is wrongly classified as AFF. Similarly, False Negative (FN) is the set of AFF emails that is wrongly classified as legitimate. Table 2 summarizes the result of calculating FP and FN rates for all classifiers. This result shows that SVM outperforms all classifiers because it has lowest false positive of 0.0435 of legitimate emails being classified as AFF.

Table 2. FP and FN rates for all classifiers

Classifier	FP	FN
BNN	0.0565	0.0282
SVM	0.0435	0.0282
BLR	0.0487	0.0310
NBC	0.0742	0.0679

The precision, recall and F-measure are also determined to compare classification accuracies of the classifiers. Precision measures the validity ratio to distinguish AFF emails predicted as positive. Recall measures the fraction of all AFF emails classified to reflect the performance that AFF emails are successfully distinguished. F-measure is generally defined as the harmonic mean of precision and recall measures. Table 3 summarises this result wherein SVM is seen to have the highest precision of about 98% and F-measure of about 98%.

Table 3. Comparison of Precision, Recall and F-measures for all classifiers

Classifier	Precision	Recall	F-measures
BNN	0.9746	0.9725	0.9736
SVM	0.9803	0.9718	0.9760
BLR	0.9690	0.9780	0.9734
NBC	0.9655	0.9320	0.9485

In this study, we also compare classification accuracies of classifiers. Accuracy is the fraction of emails (AFF and legitimate) correctly predicted by a classifier relative to the size of the dataset. Table 4 summaries this result when cross-validation was used and this shows that SVM has the highest accuracy of about 96.43%.

Table 4. Accuracy rates for all classifiers

Classifier	TP	TN	Accuracy
BNN	0.9725	0.9435	0.9580
SVM	0.9720	0.9565	0.9643
BLR	0.9690	0.9512	0.9601
NBC	0.9320	0.9289	0.9289

Receiver Operating Characteristic (ROC) curve is an important measure used to compare classification accuracies of classifiers. ROC curve is a plot of True Positive Rate (TPR) against False Positive Rate (FPR) for diverse possible cut-points of an accuracy test. We estimate the Area Under ROC Curve (AUC) as a measure of classification accuracy, wherein an area of 1 and 0 represent best and worst accuracies respectively. AUC is defined as isocost gradient chosen as tangent point on the highest isocost line that touches the curve. Table 5 shows AUC computation for all classifiers and SVM is seen to give the best accuracy because it gives the highest AUC of 96.45%. Judging from results presented in Tables 1-5, SVM is nominated as the most effective classifier that can assist to detect AFF activities on internet.

Table 5. Area under the ROC curve (AUC) for all classifiers

Classifier	Cut-point	TPR	FPR	AUC
BNN	100	0.05483	0.03043	0.9573
SVM	100	0.04193	0.02898	0.9645
BLR	100	0.05000	0.03043	0.9597
NBC	100	0.07419	0.06739	0.9292

6 Conclusion

In this study, we compare classification accuracies of four supervised learning classifiers to determine the one that can assist us to accurately detect AFF activities on internet. The natural language processing methodology of removing noisy and stop words, stemming and clustering semantically related words is used. Cross-validation, recall, precision, F-measure, AUC and cost sensitive analysis and weighted accuracy are used to compare classification accuracies of classifiers. The introduction of globalCM algorithm not only reduces dimension, but also overcomes sparseness problems and improves computational efficiency.

The results of experiments conducted in this study to compare classification accuracies of classifiers show that SVM outperforms all other classifiers, making it more appealing to detect AFF activities on internet. Moreover high levels of results (Tables 1-5) obtained for all classifiers give an indication that cluster features give an effective representation that characterizes AFF activities on internet. Consequently, objectives of this study are met. In the current study, cluster size is chosen arbitrary, but in future work it will be varied to determine the effect of cluster sizes on classification accuracies. In future, we also intend to combine social networks analysis to gain more insight on traffic flow of AFF defrauders in all geographical locations. This will provide a better understanding of how to build rich sources of learning about cybercrime activities on internet.

References

1. Grobier, M.: Strategic information security: facing the cyber impact. In: Proceedings of the Workshop on ICT uses in Warfare and Safeguarding of Peace, pp. 12–22. SAICSIT (2010)
2. Internet Crime Complaint Center (IC3). An FBI–NW3C partnership, <http://www.ic3.gov/media/annualreports.aspx> (accessed July 2011)
3. UAGI. Ultrascan 419unit-419 Advance Fee Fraud Statistics, http://www.ultrascanagi.com/public_html/html/pdf_files/419_Advance_Fee_Fraud_Statistics_2009.pdf
4. Marcus, K.R., Seigfried, K.: The future of computer forensics:a needs analysis survey. *Computer & Security* 23(1), 12–16 (2004)
5. Ciardhuáin, O.S.: An extended model of cybercrime investigations. *International Journal of Digital Evidence* 3(1) (2004)

6. Chandrasekaran, M., Narayanan, K., Upadhyaya, K.S.: Phishing email detection based on structural properties. In: *First Annual Symposium on Information Assurance: Intrusion Detection and Prevention*, New York, pp. 2–8 (2006)
7. Abu-Nimeh, S., Nappa, D., Wang, X., Nair, S.: A comparison of machine learning techniques for phishing detection. In: *Proceedings of the Anti-Phishing Working Groups (APWG), Second Annual eCrime Researchers Summit*, Pittsburgh, PA, US, pp. 1–10 (2007)
8. Fette, I., Sadeh, N., Tomasic, A.: Learning to detect phishing emails. In: *Proceedings of the 16th International Conference on World Wide Web*, pp. 649–656. ACM Press, New York (2007)
9. Airoidi, E., Malin, B.: Data mining challenges for electronic safety: the case of fraudulent intent detection in emails. In: *Proceedings of the Workshop on Privacy and Security Aspects of Data Mining, IEEE International Conference on Data Mining*, Brighton, England, pp. 1–10 (2004)
10. Hadjidj, R., Debbabi, M., Lounis, H., Iqbal, F.: Towards an Integrated Email Forensic Analysis Framework. *Digital Investigation* 5, 124–137 (2009)
11. Modupe, A., Olugbara, O.O., Ojo, S.O.: Identifying advanced fee fraud activities on the internet using machine learning algorithms. In: *3rd IEEE International Conference on Computational Intelligence and Industrial Application (PACIIA)*, Wuhan, China, pp. 240–242 (2010)
12. Wenliang, C., Xingzhi, C., Huizhen, W., Jingbo, Z., Tianshun, Y.: Automatic word clustering for text categorization using global information. In: *AIRS*, Beijing, China, pp. 1–6. ACM (2004)
13. Worth, A.P., Cronin, M.T.D.: The use of discriminant analysis, logistic regression and classification tree analysis in the development of classification models for human health effects. *Journal of Molecular Structure* 622, 97–111 (2003)
14. Khan, A., Baharudin, B., Lee, L.H., Khan, K.: A review of machine learning algorithms for text documents classification. *Journal of Advanced in Information Technology* 1(1), 4–20 (2010)
15. Byvatov, E., Fechner, U., Sadowski, J., Schneider, G.: Comparison of support vector machine and artificial neural network systems for drug/nondrug classification. *J. Chem. Inf. Comput. Sci.* 43, 1882–1889 (2003)
16. Yu, B., Xu, Z., Li, C.: Latent semantic analysis for text categorization using neural network. *Knowledge-Based Systems* 24, 900–904 (2008)
17. Bishop, C.M.: *Neural networks for pattern recognition*. Oxford University Press (1995)
18. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern classification*. Wiley-Interscience, New York (2000)
19. Cortes, C., Vapnik, V.: *Support vector networks in machine learning*, vol. 20, pp. 273–297 (1995)
20. Rios, G., Zhu, H.: Exploring support vector machines and random forests for spam detection. In: *Proceedings of CEAS 2004* (2004)
21. Mitra, V., Wang, C., Banerjee, S.: Text classification: a least square support vector machine approach. *Applied Soft Computing* 7, 908–914 (2007)
22. Porter, M.: An algorithm for suffix stripping. *Program* 14(3), 130–137 (1980)
23. Kurz, T., Stoffel, K.: Going beyond stemming: creating concept signatures of complex medical terms. *Knowledge Based Systems* 15, 309–313 (2002)

24. Klimt, B., Yang, Y.: The Enron Corpus: A New Dataset for Email Classification Research. In: Boulicaut, J.-F., Esposito, F., Giannotti, F., Pedreschi, D. (eds.) ECML 2004. LNCS (LNAI), vol. 3201, pp. 217–226. Springer, Heidelberg (2004)
25. Salton, G., Yang, C., Wang, A.: A vector space model for automatic indexing. *Communications of the ACM* 18(11), 613–620 (1975)
26. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software. *SIGKDD Explorations* 11(1) (2009)
27. Wang, T., Chiang, H.: Fuzzy support vector machine for multi-class text categorization. *Information Process and Management* 43, 914–929 (2007)