Neural Comput & Applic (2019) 31:691–699
DOI 10.1007/s00521-017-3100-y

ORIGINAL ARTICLE

Improved email spam detection model based on support vector machines

Sunday Olusanya Olatunji[1]

Received: 15 October 2016 / Accepted: 15 June 2017 / Published online: 28 June 2017

Abstract Email has become extremely popular among people nowadays. In fact, it has been reported to be the cheapest, popular and fastest means of communication in recent times. Despite the huge benefits of emails, unfortunately its usage has been bedeviled with the huge presence of unsolicited and sometimes fraudulent emails which must be promptly detected and isolated through what is popularly referred to as spam detection system. Spam detection is highly needed to protect email users and prevents several negative usages to which emails have been subjected to of recent. Unfortunately, due to the adaptive nature of unsolicited emails through the use of mailing tools, the effectiveness of the spam detecting tools has often been limited and sometimes rendered ineffective, hence the need for better spam detection tools to achieve better spam detection accuracy. Several spam detection models have been proposed and tested in the literature, but still the reported accuracy indicated that there is still need for more work in this direction in order to achieve better accuracy. In this work, support vector machines-based model is proposed for spam detection while paying attention to appropriately search for the optimal parameters to achieve better performance. Experimental results show that the proposed model outperformed all the earlier proposed models on the same popular dataset employed in this work. Accuracy of 95.87 and 94.06% was obtained for training and testing sets, respectively. The 94.06% testing accuracy represents an improvement of 3.11% over the best reported model in the literature that had an accuracy of 91.22% on the same dataset used in this work.

Keywords Support vector machines · Email · Spam · Non-spam · Spam detector · Computational intelligence

1 Introduction

Email has become extremely popular among people nowadays. In fact, people can hardly do without sending or receiving email messages on daily basis. The use to which email has been put into includes but not limited to communicating important messages within an organization, inter-organization or worldwide; advertisements, job recruitment processing and communicating between countries. In essence, the importance of email in our present day life cannot be overemphasized. Despite the huge benefits of emails, unfortunately its usage has been bedeviled with the huge presence of unsolicited and sometimes fraudulent emails which must be promptly detected and isolated through what is popularly referred to as spam detection system. Spam detection is used to differentiate between spam email and non-spam emails, thereby making it possible to prevent spam mail from getting into the inbox of users. Thus, it could be stated that spam detection is the first step and the most important stage in the email filtering process toward ensuring that junk mails are prevented from entering users' inbox, particularly in this age of huge spam messages due to the availability of bulk mailing tools that have pushed up the amount of spam emails in a skyrocketed manner.

It is a fact that spam detection is highly needed to protect email users and prevents several negative usages to which emails have been subjected to of recent.

✉ Sunday Olusanya Olatunji
oluolatunji.aadam@gmail.com; osunday@uod.edu.sa

[1] Computer Science Department, College of Computer Sciences and Information Technology, University of Dammam, Dammam, Kingdom of Saudi Arabia

Unfortunately, due to the adaptive nature of unsolicited emails through the use of mailing tools, the effectiveness of the spam detecting tools has often been limited and sometimes rendered ineffective or compromised, hence the need for better spam detection tools to achieve better spam detection accuracy. Several spam detection models have been proposed and tested in the literature, but still the reported accuracy still begs for more work in this direction in order to achieve better accuracy. Authors in Özgür et al. [13] made use of artificial neural network-based model for spam detection but only succeeded in achieving 86% accuracy which is still considered far from the ideal. In Zhang et al. [36], the authors applied naive Bayes approach while incorporating cost-sensitive multi-objective genetic programming for feature extraction and used it for spam detection but achieved an accuracy of 79.3% correctly detected email types. Moreover, in Ariaeinejad and Sadeghian [7], the authors presented a spam detection system based on interval type-2 fuzzy sets, exploring the capabilities of fuzzy logic of type-2 category, but could only succeed in obtaining spam detection accuracy of 86.9% for the testing set. The authors in Temitayo et al. [33] made use of genetic algorithm-based hybrid on the same dataset but were only able to push the accuracy to 90% for the testing set. Furthermore and of recent is the work of Idris and Selamat [12], where a hybrid model consisting of smart hybridization of negative selection algorithm (NSA) with particle swarm optimization (PSO) was presented. They were able to push the accuracy to 91.22% for the testing set. Considering the work done so far and the performance accuracy obtained till date, it is clear that there is need to further explore the possibility of achieving better results using the same popular datasets. This work is thus set to come up with alternative model that could push the accuracy level higher than previous models.

In this paper, a support vector machines (SVM)-based model is proposed and investigated toward achieving better accuracy of spam detection while paying strong attention to appropriately using exhaustive parameter search techniques to ensure better spam detection accuracy. SVM has become the target of several researchers of recent due to its ability to appropriately generalize in the face of either few data samples or huge data samples. This technically means that it does not suffer from the problem of dimensionality [20, 34], which is a general problem that often confronts other machine learning techniques. SVM has its theoretical basis in statistical leaning theory by deploying its powerful phenomenon otherwise known as kernel trick. This kernel trick made it possible to project non-separable problems into higher-dimensional space where it will become easily separable using any of the various available kernel functions [8, 23]. The unique capability of SVM has often made it a preferred choice in solving unique problems as in Rojas et al. [30], Yin et al. [35] and others.

SVM has been chosen for this work partly following its numerous successful applications into various fields often with excellent performance and unique generalization ability. Some of the recent onslaughts to which SVM has been put to use of recent including those by the author include but not limited to application to the field of biomedical engineering, materials properties modeling, oil and gas properties modeling, stock price forecasting and host of others [2–6, 9–11, 16, 17, 19–29, 32].

In this work, SVM has been utilized in a MATLAB environment for the experiment. In addition, a systematic parameter search algorithm presented in Sect. 3 has been utilized to facilitate achieving better classification accuracy by SVM since the algorithm ensures that optimal values are obtained for the entire SVM parameters. Experimental results show that the proposed model outperformed all the earlier proposed models on the same popular dataset employed in this work. Accuracy of 95.87 and 94.06% was obtained for training and testing sets, respectively. The 94.06% testing accuracy represents an improvement of 3.11% over the best reported model in the literature which had an accuracy of 91.22%. Also other quality measure investigated shows that the proposed model prediction was not random but rather achieved excellent measures required of viable classification models. With the excellent accuracy achieved in this work, SVM has again proven to be a very reliable and promising computational intelligence technique to resort to in achieving excellent prediction accuracy.

The remaining part of this work is organized as follows. Section 2 contains the proposed model. Section 3 contains empirical studies that include dataset description, experimental setup or methodology, and the adopted optimization strategy or parameter search strategy. Section 4 presents results and discussion, while Sect. 5 contains the conclusion and recommendation emanating from this work.

2 Proposed technique

SVM is a statistical-based machine learning technique with unique ability to model complex relationships among variables [8]. It uniquely combines generalization control with a technique to address the curse of dimensionality. Curse of dimensionality often limits the performance of machine learning techniques in the face of few data simples, but for support vector machines it has distinguished itself as a unique technique with ability to perform excellently even in the face of few data samples [2, 8, 20, 25]. In support vector machines, the formulation results in a global quadratic optimization problem with box constraints, which is readily solved by interior point methods [8, 34]. Support vector machines are uniquely empowered through its kernel functions to easily map non-separable problems to higher dimensions where they become easily separable. The kernel mapping provides a unifying framework for most of the commonly employed model architectures, enabling comparisons to be performed [2, 8, 34]. Kernel mapping makes non-separable cases to become separable by transforming the original dataset from the initial dimensional space to another higher-dimensional space where separation becomes possible as can be seen in Fig. 1.

SVM, which was primarily developed for classification problems, has also been extended to regression problems. In classification problems, generalization control is obtained by maximizing the margin, which corresponds to minimizing the weight vector in a canonical framework [34]. The solution is obtained as a set of support vectors that can be sparse. These lie on the boundary and as such summarize the information required to separate the data and the possible unseen data samples later. Figure 2 shows how a margin is created between two sets of data in a classification problem using support vector machines.

2.1 Further details on the proposed model

Generally, in prediction and classification problems, the purpose is to determine the relationship among the set of input and output variables of a given dataset $D = \{Y, X\}$ where $X \in R^p$ represents the n-by-p matrix of p input variables also know as predictors or independent variables. It may be noted that $Y \in R$ for forecasting or regression problems and $Y \subseteq R$ for classification problems. Now for the case of classification as we have in this work, Suppose $D = \{y_i, x_{i1}, \ldots, x_{ip}\}$ is a training set for all $i = 1, \ldots, n$ of input variables X_j where $[X_j = (x_{1j}, \ldots x_{nj})^{\mathrm{T}}]$ for $j = 1, \ldots, p$, and the output variables, $Y = (y_1 \ldots y_n)^{\mathrm{T}}$. The lower case letters $x_{i1}, x_{i2}, \ldots, x_{ip}$ for all $i = 1, \ldots, n$ refer to the values of each observation of the input variables, and $y = k$ to the response variable Y to refer to class A_k for all $k = 1, 2, \ldots, c$, where $c \geq 2$, but in this case of spam detection, $c = 2$ representing spam or non-spam two-class labels.

In what follows, the basic ideas behind SVM for pattern recognition, especially for the two-class classification problem, are briefly described and readers are referred to Cortes and Vapnik [8] and Vapnik [34] for a full description of the technique.

According to Cortes and Vapnik [8] and Vapnik [34], the goal of two-class SVM is to construct a binary classifier or derive a decision function from the available samples which has a small probability of misclassifying a future sample. The proposed SVM implements the following idea: It maps the input vectors $\overrightarrow{x} \in R^d$ into a high-dimensional feature space $\Phi(\overrightarrow{x}) \in \mathrm{H}$ (see Fig. 1) and constructs an optimal separating hyperplane (OSH), which maximizes the margin, which is the distance between the hyper plane and the nearest data points of each class in the space H (see Fig. 2). 1

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

where b is the bias and the coefficients α_i are obtained by solving the following convex quadratic programming (QP) problem:

$$\begin{aligned} &\text{Maximize} \quad \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j \cdot y_i y_j \cdot K(\vec{x_i}, \vec{x_j}) \\ &\text{Subject to} \quad 0 \leq \alpha_i \leq C \qquad (2) \\ &\sum_{i=1}^{N} \alpha_i y_i = 0 \quad i = 1, 2, \ldots n. \end{aligned}$$

In Eq. (2), C is a regularization parameter which controls the trade-off between margin and misclassification error. These x_j are called support vectors only if the corresponding $\alpha_i > 0$. Further details on the algorithmic implementation of SVM utilized in this work could be found in Canu et al. [31].

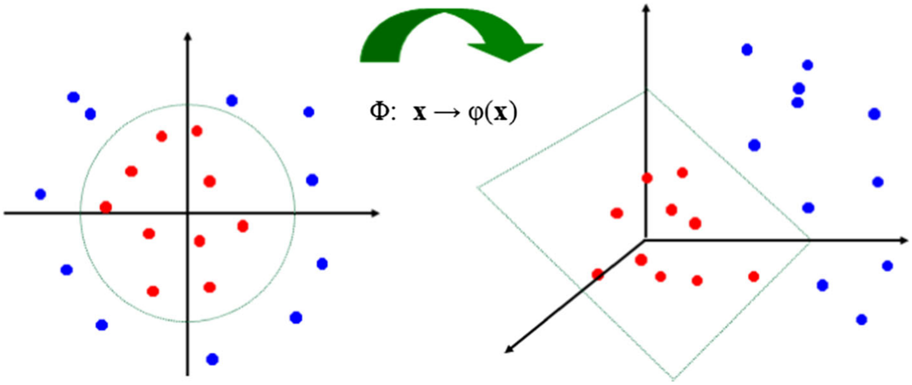
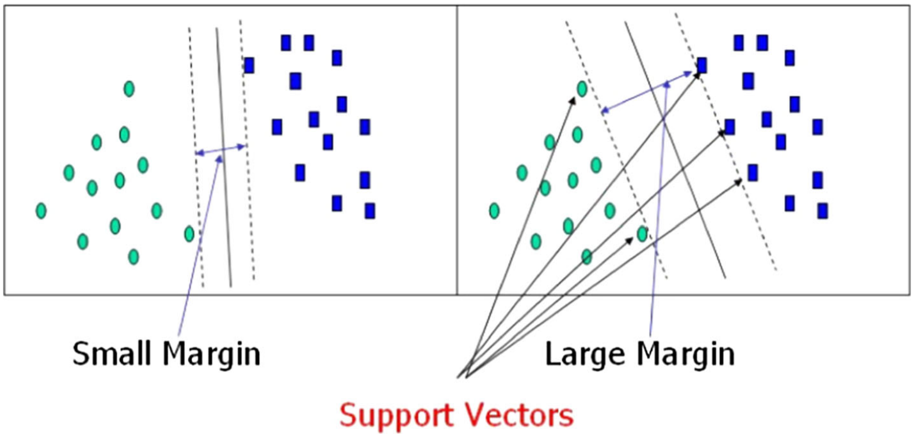


Fig. 1 Mapping to a higher dimension using kernel function [15]

Fig. 2 Creation of margins between two datasets by support vectors



Kernel function: The kernel function is responsible for transforming the dataset into hyper plane. Typical kernel functions are represented as:

$$K(\vec{x_i},\vec{x_j}) = (\vec{x_i}\cdot\vec{x_j}+1)^d, \tag{3}$$

$$K(\vec{x_i},\vec{x_j}) = \exp\left(-\gamma\left\|\vec{x_i}-\vec{x_j}\right\|^d\right), \tag{4}$$

Equation (3) is the polynomial kernel function of degree d which will revert to the linear function when d = 1. Equation (4) is the radial basis function (RBF) kernel with one parameter γ otherwise referred to as Gaussian kernel function (Mahmoud and Olatunji 2009; Olatunji 2011; [34].

Other kernel functions are:

$$\begin{aligned}&\text{Linear}:\\&K(x_i,x_j)=x_i^{\mathrm{T}}x_j\,\text{and sigmoid}:\;K(x_i,x_j)\\&\qquad=\tan h(\gamma x_i^{\mathrm{T}}x_j+r)\end{aligned}$$

where $\gamma, r,$ and d are kernel parameters.

Regularization parameters (C): This determines the trade-off cost between minimizing the training error and minimizing the model's complexity.

3 Empirical studies and computational methodologies

For the empirical work, the popular and earlier used dataset [14] by several researchers was acquired. Computational intelligence methodologies and procedures based on SVM are then followed to arrive at the final outcome of the empirical work. This was started with the description of dataset acquired, then followed by performance measure criteria, description of experimental settings and implementation, and finally, the parameter search or optimization procedures are presented.

3.1 Dataset description

The dataset utilized in this work is the popular and often used corpus benchmark spam dataset that is acquired from email spam messages [14]. The dataset is made of 57 features (predictors attributes) and 1 target attributes, which is the class label in the corpus indicating the status of each dataset sample or instance as either being a spam or non-spam, represented by 1 or 0, respectively. Further information regarding the dataset could be found in Hopkins et al. [14].

3.2 Performance measure criteria for the proposed model

The need to have acceptable performance measure criteria for any proposed model cannot be overemphasized since it is the basis of determining how promising or otherwise a proposed model is. In this research, the popular and commonly used performance measures for the classification problems shall be employed and they include: specificity (SP), sensitivity (SN) and of course the accuracy, which is the overall percentage of correctly classified samples. Brief description of these performance measures is given below.

(i) Specificity (SP): This is a measure of the proportion of negative patterns that are correctly classified or recognized as negative. It is represented by the equation:

$$\mathrm{SP}=\frac{\mathrm{TN}}{\mathrm{TN}+\mathrm{FP}} \tag{5}$$

(ii) Sensitivity (SN): This is a measure of the proportion of positive patterns that are correctly classified or recognized as positive. It is represented by the equation:

$$\mathrm{SN}=\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}} \tag{6}$$

(iii) Accuracy (Acc): It is the overall percentage of samples correctly classified. It is represented by the equation:

$$\mathrm{Acc} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{7}$$

where TP is true positive, TN is true negative, FP is false positive, and FN is false negative.

3.3 Experimental settings and implementation

The experimental procedures implemented here followed strictly the computational intelligence approach. The acquired dataset was first divided into training and testing sets in the ratio 7:3, respectively, using the stratified sampling approach. Then, the training dataset was first introduced to the models for training and validation. Thereafter, the reserved 30% of the dataset was then used to test the system in order to ascertain the performance accuracy of the proposed model.

It must be noted at this point that the implementation of the model was carried out using written codes in MATLAB 2012b environment, although some SVM implementation functions from SVM toolbox in Canu et al. [31] were made use of where necessary. The implementation proceeded by providing the training set to the system to train and generate necessary support vectors that will be used during the testing face to achieve the needed identification of the mail messages as either spam or non-spam. After successfully training the SVM model and validation, the testing set that has been kept away from the system is then presented to be used for the actual testing that will determine the performance accuracy of the proposed system.

It must be stated at this point that, to achieve any meaningful training and subsequent testing of any predictive classification model, the need to search for and arrive at the best possible values for its parameters (optimal parameters values) cannot be overemphasized. Efforts have been made in this work to ensure systematic parameter search that will provide better classification accuracy. Since SVM classification has very important parameters that often affect and determine its predictive ability depending on the problem at hand, a systematic parameter search approach has been used in this work to ensure that the best optimal parameter values were achieved. This parameter search procedure is described as follows.

3.3.1 *Optimal parameters search procedure implemented for the proposed SVM spam detector (adapted with modifications from Olatunji et al. [18])*

The parameters associated with the SVM were optimized through a test-set-cross-validation on the available dataset. The procedure goes thus: For each run of generated training and testing sets, the values of the accuracy (percentage of correctly classified samples) were monitored for a group of parameters C (bound on the Lagrangian multiplier) and λ (conditioning parameter for QP methods). Searching through all possible values of the parameters in a given range will identify the best performance measures and the corresponding values of the parameters for the fixed set of features. In the experiments, this process was repeated for every SVM kernel option available, each time with an incremental step of parameters. The optimal values of the parameters and the kernel option associated with the best performance measure were identified. A summary of the procedure is as follows:

Step 1: Choose the initial "kernel" option from the list of available kernel options (Gaussian, Polynomial, etc.).

Step 2: Identify the best values of the parameters C and ε through a test-set cross-validation and store the corresponding performance measure (i.e., accuracy).

Step 3: If there is no kernel option left, then go to step 4. Otherwise, add the next kernel option and go to step 2.

Step 4: Identify the best performance measure and its associated kernel option and the parameters values.

Step 5: Use the optimized kernel option and the corresponding parameters values to train the final model.

Step 6: Compute the performance measures for both the training and testing sets using the system obtained in step 5.

The above-stated procedures could be presented algorithmically as follows:

Let the set A contain all the possible kernel options, the element of A is of the form $A_i(j)$, where i is the kernel function number, j is the index for selected values of C, k is the index for selected values of ε, nf is the total number of kernel functions available or selected, nc is the maximum value of C assumed, and $n\lambda$ is the maximum value of ε assumed. Also pm represents performance measure taken, ix represents index for best kernel function, jx represents index for best value of C, and kx represents index for best value of ε. The algorithm then goes thus:

Algorithm 1.0: Optimal parameters search procedure for the proposed SVM SPAM detector *(adapted from* (Olatunji et al., 2014) *)*

```
Initialization; jx=0, vx=0, ix=0, kx=0
for $i = 1 \rightarrow nf$
for $j = 1 \rightarrow nc$
  for $k = 1 \rightarrow n\lambda$
    $pm = f(A_i(j))$ {Performance measure for the present parameters combination}
    if $pm$ is better than $vx$ then $vx = pm$
    $ix = i$ ; $jx = j$ ; $kx = k$ {storing the index of the better parameter}
  end
 end
end
```

Table 1 Optimum parameters for the proposed SVM email spam detection model

C	3
Hyper-parameter(Lambda)	0.0001
Epsilon(ε)	2^{-1}
Kernel	Gaussian

With the above-presented algorithmic system for the proposed spam detection model, the best model parameter selection is thus assured or made possible through the exhaustive search approach used. This invariably led to the achievement of better performance recorded by the proposed model.

Based on the above-described algorithmic search procedures, the following optimum parameter values were obtained and used for the final model development (Table 1).

3.3.2 *Further description of the model implementation procedures*

The developed spam detector followed strictly the computational intelligence approach to classifier implementation processes. To further clarify the internal data flow through the entire system, a data flow diagram has been provided to this end. The following figure depicts the data flow diagram of proposed SVM-based improved spam detection model.

It must be stated here that the proposed model followed strictly the standard training and testing procedures in which the testing set is kept away as unseen data before it is sent to the model for testing after successful training must have been concluded. From the data flow diagram depicted in Fig. 3, it is made clear how the dataset is first partitioned into training set and testing set. Having trained the SVM model with the training dataset, then the system proceeded to the testing process as follows. As shown in Fig. 3, the testing set is separated into the attributes and the target (actual label, either spam or non-spam). The testing set attributes are first sent into trained system for testing, and the SVM-based spam detector generates its final predicted output (spam or non-spam). The predicted outputs are then compared to the preserved target (actual known labels) in order to determine the accuracy of the model during testing. With the completion of this stage, the proposed spam detector is finally ready to identify any provided email attributes as either spam or non-spam.

4 Experimental results and discussion

Experimental results from simulations indicated that the proposed model shows an improved accuracy when compared with earlier used models on the same dataset.

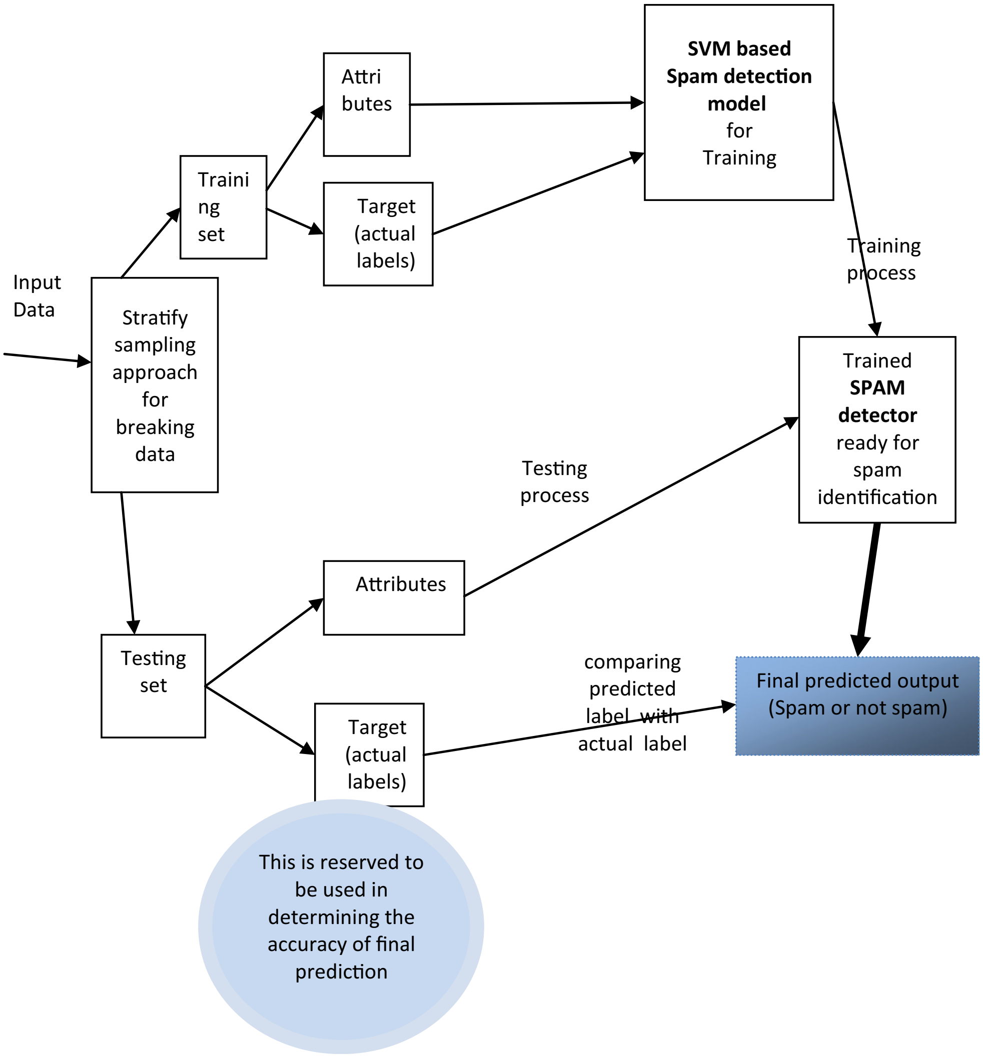

SVM based Spam detection model for Training
Attri butes
Traini ng set
Target (actual labels)
Input Data
Stratify sampling approach for breaking data
Training process
Trained SPAM detector ready for spam identification
Testing process
Attributes
Testing set
comparing predicted label with actual label
Final predicted output (Spam or not spam)
Target (actual labels)
This is reserved to be used in determining the accuracy of final prediction

Fig. 3 Data flow diagram of the proposed SVM-based spam detector

Table 2 (a) Confusion matrix for the training set; (b) confusion matrix for the testing set (Note: in the confusion matrix below, *1* stands for *spam* email while *0* stands for *not spam*)

(a)

	0	1
0	1927	83
1	50	1161

(b)

	0	1
0	777	48
1	34	521

Performance measures outcomes for the proposed SVM-based spam detection system is first presented below and thereafter, it is followed by the comparison of the accuracy of the proposed model with those of earlier used models on the same dataset.

The confusion matrix for the proposed model is presented in Table 2 for both training and testing phases. From the confusion matrix, the essential performance measures that include specificity, sensitivity and accuracy were computed using respective standard performance measure formula earlier discussed in Sect. 3.

From the confusion matrices above, the quality measures are calculated as follows based on the details earlier provided in Sect. 3.2:

For the training set, given that from Table 2(a), TP = 1161, FP = 83, TN = 1927, FN = 50, we have:

i. Specificity (SP): $SP = \frac{TN}{TN+FP} = 0.958706$

ii. Sensitivity (SN): $SN = \frac{TP}{TP+FN} = 0.958712$

iii. Accuracy (Acc): $Acc = \frac{TP+TN}{TP+TN+FP+FN} = 0.958708$

For the testing set, given that from Table 2(b), TP = 521, FP = 48, TN = 777, FN = 34, we have:

i. Specificity (SP): $SP = \frac{TN}{TN+FP} = 0.941818$

ii. Sensitivity (SN): $SN = \frac{TP}{TP+FN} = 0.938739$

iii. Accuracy (Acc): $Acc = \frac{TP+TN}{TP+TN+FP+FN} = 0.94058$

Table 3 Summary of results in percentage for both training and testing sets

	SP	SN	Acc
Training	95.87	95.87	95.87
Testing	94.18	93.87	94.06

Table 4 Comparison of testing accuracy for the proposed SVM-based spam detector and other earlier published classifiers on the same dataset

Classifiers	Accuracy (%)
Proposed SVM-based spam detector	94.06
NSA–PSO [12]	91.22
PSO [12]	81.32
NSA [12]	68.86
BART [1]	79.3
IT2 fuzzy set [7]	86.9

Finally, the overall result for both training and testing sets is summarized in Table 3.

From the above results, it is clear that the proposed spam detector achieves excellent results requisite of any preferred formidable model. In order to appreciate and make the improvement provided by this proposed model clearer, its accuracy compared to other earlier schemes is presented below.

From the results in Table 4, it could be clearly seen that the proposed SVM-based spam detector has outperformed earlier used classifiers on the same dataset [12]. It could be noticed that the proposed SVM spam detector achieved an improvement of 3.11% over the best among the other earlier schemes, which is the hybridized negative selection algorithm–particle swarm optimization (NSA–PSO) proposed in Idris and Selamat [12]. NSA–PSO was reported to be the hybrid of NSA and PSO in order to achieve better accuracy [12], yet the proposed SVM-based classifier in this work outperformed all the three including the hybrid schemes. It also provided an accuracy improvement of 18.6% over that of Bayesian additive regression trees (BART) for spam classification proposed in Abu-Nimeh et al. [1] while providing an accuracy improvement of 8.2% over that of interval type-2 fuzzy logic-based classifier proposed in Ariaeinejad and Sadeghian [7].

Thus, it is very clear from the obtained accuracy as presented in Table 4 that the proposed SVM-based spam detector in this work outperformed earlier schemes implemented on the same dataset. This work has further corroborated the often reported superior performance of SVM models in various fields of applications [2, 11, 21, 27, 29].

5 Conclusion

In this work, an improved email spam detector based on SVM classifier has been proposed, trained and tested using popular and often used standard database. Empirical results from simulation indicated that the proposed SVM-based scheme outperformed other recently published spam detector schemes tested on the same popular database used in this study. The need for a better and more accurate email spam detector scheme cannot be overemphasized; hence, the proposed scheme in this research work came appropriately and timely as an improved scheme over the best among the previous methods used on the same dataset. In fact, the proposed SVM-based spam detector provides improvement of 3.11% over the NSA–PSO hybrid scheme that happens to be the best among the previous reported spam detection schemes. The success recorded in this work has further corroborated the unique reputation of SVM as viable and reliable prediction or classification tool with excellent performance in different field of applications. As a result of the promising results achieved in this work, efforts shall be made next, to investigate any possible means to improve upon the performance while also exploring the unique capability of SVM classifies in other germane areas where accurate prediction or classification outcomes are highly desirable. It is also recommended that efforts should always be made first in exploring simple standalone system appropriately before delving into hybrid scheme since it has been shown that a standalone application could easily outperform hybrid if its parameters are systematically optimized as it has been demonstrated in this work with SVM model outperforming earlier hybrid schemes of NSA–PSO.

Acknowledgement The author would like to acknowledge the University of Dammam, Dammam, Kingdom of Saudi Arabia for some of the facilities utilized during the course of this research.

Compliance with ethical standards

Conflict of interest The author declares that he has no conflict of interest.

References

1. Abu-Nimeh S, Nappa D, Wang X, Nair S (2008) Bayesian additive regression trees-based spam detection for enhanced email privacy. In: 2008 third international conference on availability, reliability and security. IEEE, pp. 1044–1051. doi:10.1109/ARES.2008.136
2. Adewumi AAAA, Owolabi TO, Alade IOIO, Olatunji SO (2016) Estimation of physical, mechanical and hydrological properties of permeable concrete using computational intelligence approach. Appl Soft Comput 42:342–350. doi:10.1016/j.asoc.2016.02.009

3. Akande KOKO, Owolabi TO, Olatunji SO (2015) Investigating the effect of correlation-based feature selection on the performance of support vector machines in reservoir characterization. J Nat Gas Sci Eng 22:515–522. doi:10.1016/j.jngse.2015.01.007
4. Akande KO, Olatunji SO, Owolabi TO, AbdulRaheem A (2015a) Comparative analysis of feature selection-based machine learning techniques in reservoir characterization. CPAPER, Society of Petroleum Engineers. doi:10.2118/178006-MS
5. Akande KO, Olatunji SO, Owolabi TO, AbdulRaheem A (2015b) Feature selection-based ANN for improved characterization of carbonate reservoir. CPAPER, Society of Petroleum Engineers. doi:10.2118/178029-MS
6. Akande KO, Owolabi TO, Twaha S, Olatunji SO (2014) Performance comparison of SVM and ANN in predicting compressive strength of concrete. IOSR J Comput Eng 16(5):88–94
7. Ariaeinejad R, Sadeghian A (2011) Spam detection system: a new approach based on interval type-2 fuzzy sets. In: 2011 24th Canadian conference on electrical and computer engineering(CCECE). IEEE, pp. 000379–000384. doi:10.1109/CCECE.2011.6030477
8. Cortes C, Vapnik V (1995) Support vector networks. Mach Learn 20:273–297
9. Fernandez R, Picard RW (2002) Dialog act classification from prosodic features using support vector machines. In: Speech Prosody. Conference paper, Aix-en Provence, France, Dialog Act
10. Gupta SM (2007) Support vector machines based modelling of concrete strength. World Acad Sci Eng Technol 36:305–311
11. Ibitoye M, Hamzaid N, Abdul Wahab A, Hasnan N, Olatunji S, Davis G (2016) Estimation of electrically-evoked knee torque from mechanomyography using support vector regression. Sensors 16(7):1115. doi:10.3390/s16071115
12. Idris I, Selamat A (2014) Improved email spam detection model with negative selection algorithm and particle swarm optimization. Appl Soft Comput 22:11–27. doi:10.1016/j.asoc.2014.05.002
13. Özgür L, Güngör T, Gürgen F (2004) Spam mail detection using artificial neural network and Bayesian filter, 505–510. doi:10.1007/978-3-540-28651-6_74
14. Hopkins M, Reeber E, Forman G, Suermondt J (1999) SpamBase dataset. Hewlett-Packard Labs; 1501 Page Mill Rd.; Palo Alto; CA 94304. https://archive.ics.uci.edu/ml/datasets/Spambase
15. Milano P, Chicco D (2012) Support vector machines in bioinformatics: a survey. A technical report, pp 1–35. https://s3-us-west-2.amazonaws.com/mlsurveys/125.pdf. Accessed June 2017
16. Ni L-P, Ni Z-W, Gao Y-Z (2011) Stock trend prediction based on fractal feature selection and support vector machine. Expert Syst Appl 38(5):5569–5576. http://www.sciencedirect.com/science/article/B6V03-51F7PMS-B/2/f3645bc7144b2047233ac753849dccce
17. Olatunji SO, Hossain A (2012) Support vector machines based model for predicting software maintainability of object-oriented software systems. J Inf Commun Technol 2(5), 23–32. http://www.jict.co.uk/volume-2-issue-5-may-2012
18. Olatunji SO, Selamat A, Abdulraheem A, Abdul Raheem AA (2014) A hybrid model through the fusion of type-2 fuzzy logic systems, and extreme learning machines for modelling permeability prediction. Inf Fusion 16(2014):29–45. doi:10.1016/j.inffus.2012.06.001
19. Owolabi T, Akande K, Olatunji S (2014) Estimation of superconducting transition temperature T C for superconductors of the doped MgB2 system from the crystal lattice parameters using support vector regression. J Supercond Novel Magn. doi:10.1007/s10948-014-2891-7
20. Owolabi TO, Akande KO, Olatunji SO (2015) Estimation of surface energies of hexagonal close packed metals using computational intelligence technique. Appl Soft Comput 31:360–368. doi:10.1016/j.asoc.2015.03.009
21. Owolabi TO, Akande KOKO, Olatunji SO (2016) Application of computational intelligence technique for estimating superconducting transition temperature of YBCO superconductors. Appl Soft Comput 43:143–149. doi:10.1016/j.asoc.2016.02.005
22. Owolabi TO, Akande KO, Olatunji SO (2014) Estimation of the atomic radii of periodic elements using support vector machine. Int J Adv Inf Sci Technol 28(28):39–49
23. Owolabi TO, Akande KO, Olatunji SO (2014) Prediction of superconducting transition temperatures for fe-based superconductors using support vector machine. Adv Phys Theories Appl 35:12–26
24. Owolabi TO, Akande KO, Olatunji SO (2014) Support vector machines approach for estimating work function of semiconductors: addressing the limitation of metallic plasma model. Appl Phys Res 6(5):122
25. Owolabi TO, Akande KO, Olatunji SO (2015) Development and validation of surface energies estimator (SEE) using computational intelligence technique. Comput Mater Sci 101:143–151. doi:10.1016/j.commatsci.2015.01.020
26. Owolabi TO, Akande KO, Olatunji SO (2015) Estimation of surface energies of transition metal carbides using machine learning approach. Int J Mater Sci Eng. doi:10.17706/ijmse.2015.3.2.104-119
27. Owolabi TO, Akande KO, Olatunji SO (2016) Computational intelligence method of estimating solid–liquid interfacial energy of materials at their melting temperatures. J Intell Fuzzy Syst 31:519–527
28. Owolabi TO, Akande KO, Sunday OO (2015) Modeling of average surface energy estimator using computational intelligence technique. Multidiscip Modell Mater Struct 11(2):284–296. doi:10.1108/MMMS-12-2014-0059
29. Owolabi TO, Faiz M, Olatunji SO, Popoola IK (2016) Computational intelligence method of determining the energy band gap of doped ZnO semiconductor. Mater Des 101:277–284. doi:10.1016/j.matdes.2016.03.116
30. Rojas DA, Ramos OL, Saby JE (2016) Recognition of Spanish vowels through imagined speech by using spectral analysis and SVM. J Inf Hiding Multimed Signal Process 7(4):889–897. http://bit.kuas.edu.tw/~jihmsp/2016/vol7/JIH-MSP-2016-04-020.pdf
31. Canu S, Grandvalet Y, Guigue V, Rakotomamonjy A (2008) SVM and kernel methods matlab toolbox. A free SVM toolbox. http://asi.insa-rouen.fr/enseignants/~arakoto/toolbox/. Accessed June 2017
32. Olatunji SO, Arif H (2015) Identification of erythemato-squamous skin diseases using support vector machines and extreme learning machines: a comparative study towards effective diagnosis. Trans Mach Learn Artif Intell 2(6):124–135. doi:10.14738/tmlai.26.812
33. Temitayo F, Stephen O, Abimbola A (2012) Hybrid GA-SVM for efficient feature selection in E-mail classification. ISSN 3(3):2222–1719. www.iiste.org
34. Vapnik V (1995) The nature of statistical learning theory. Springer, New York
35. Yin H, Qiao J, Fu P, Xia X (2014) Face feature selection with binary particle swarm optimization and support vector machine. J Inf Hiding Multimed Signal Process 5(4):731–739. http://bit.kuas.edu.tw/~jihmsp/2014/vol5/JIH-MSP-2014-04-014.pdf
36. Zhang Y, Li H, Niranjan M, Rockett P (2008) Applying cost-sensitive multiobjective genetic programming to feature extraction for spam e-mail filtering. Springer, Berlin, pp. 325–336. doi:10.1007/978-3-540-78671-9_28