Indonesian Journal of Electrical Engineering and Computer Science
Vol. 19, No. 1, July 2020, pp. 535~542
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v19.i1.pp535-542

Improving spam email detection using hybrid feature selection and sequential minimal optimisation

Ahmed Al-Ajeli, Raaid Alubady, Eman S. Al-Shamery
College of Information Technology, University of Babylon, Iraq

Article Info

Article history:

Received Oct 29, 2019
Revised Jan 4, 2020
Accepted Jan 19, 2020

Keywords:

E-mail spam
Feature selection
Machine learning
Sequential minimal optimisation

ABSTRACT

Communication by email is counted as a popular manner through which users can exchange in-formation. The email could be abused by spammers to spread suspicious content to the Internet users. Thus, the need to an effective way to detect spam emails are becoming clear to keep this information safe from malicious access. Many methods have been developed to address such a problem. In this paper, a machine learning technique is applied to detect spam emails. In this technique, a detection system based on sequential minimal optimization (SMO) is built to classify emails into two categories: spam and non-spam (ham). Each email is represented by a set of features extracted from its textual content. A hybrid feature selection is developed to choose a subset of these features based on their importance in process of the detection. This subset is then input into the SMO algorithm to make the detection decision. The use of such a technique provides an efficient protective mechanism to control spams. The experimental results show that the performance of the proposed method is promising compared with the existing methods.

Corresponding Author:

Ahmed Al-Ajeli,
College of Information Technology,
University of Babylon, Babylon, Iraq
E-mail: a.alajeli@itnet.uobabylon.edu.iq

1. INTRODUCTION

The range of services that can be accessed via Internet grow rapidly on daily basis. This requires providing a certain level of security against the malicious behaviour which is gradually becoming a real threat. With the fast progress in technology and applications, the need to provide tools to face the threat is becoming clear. The security issues such as worms and viruses have been regarded the main challenges by Information Technology world. Additionally, spams in the form of emails creates another dimension of threat. These spams have inappropriate content and are received by unknown senders. Receiving such emails at high rate on daily basis annoy the user. In addition, the computational resources are consumed causing degradation in the value of email service. The problem of spam email detection is described in the following. Given a set of emails partitioned to two sets (classes): ham and spam, where each email has a label, e.g. +1 for spam and 1 for ham. The goal to address such a problem is to build a detector (classifier) such that for any unseen received email, a label (also called target class) is given to that email.

A variety of approaches has been proposed to address the problem of spam email detection. The performance of most popular approaches will be briefly reviewed and critically assessed for solving the problem in hand. Clustering techniques have been applied for the problem in which the unsupervised learning were adopted [1-5]. Starting from an unlabeled dataset, the instances of the set are grouped (clustered) into two clusters: spam and non-spam. A similarity measure is applied to identify the instances in each group. In [2, 4], k-nearest neighbours (kNN) approach has been adopted. This approach produce no classification model and the classification is carried out by determining the similarity between the testing

email and the k training neighbours. Thus, use of the kNN notion causes a high cost in terms of time and memory. Generally speaking, the clustering techniques raise a difficulty to evaluate the results of the learning algorithms. As a result, this problem limits their use for applications in which it is not possible to have labelled data set.

Unlike clustering techniques, the statistical methods such as Naive Bayes method have also been applied [6-11]. In these methods, a supervised learning based on a probabilistic model have been presented. The notion used in such methods relies on Bayesian theorem in order to build a model which is able to distinguish between spam and ham emails. In its simple form, a set of statistics is collected from the dataset provided with a prior knowledge about each instance's label (class).

Another group of publications in the context of supervised learning have adopted support vector machine (SVM) notion for spam email detection [12-16]. This notion addresses the problem by formulating the spam detection as an optimastion problem to produce a prediction model in which the separation between the two classes of spam and ham is maximised. This method proved the ability to solve complex and large classification problems. For more details about more related work in this field, we refer the reader to [17].

Although many methods presented for spam email detection, yet no method can handle the problem completely. The difficulty is arisen because the type and content of spam emails constantly change over the time. In this paper, the aim is to address the problem of spam email detection based on the textual content of each email. For this end, two objectives are followed. First, a hybrid feature selection method is proposed to reduce the dimensionality and select features which are more relevant. This hybrid method is built based on two existing methods: correlation and gain ratio. As a result, a simplified detection model will be obtained. Secondly, a study about the application of the new SVM learning algorithm called sequential minimal optimisation (SMO) [18-20] is given by comparing its performance against four other common detection algorithms. In fact, the SMO algorithm is originally introduced to reduce the high computational requirements by the standard SVM, in addition it is more scalable. This gives the algorithm the capability to deal with large datasets. Here, we are in particular interested in exploring how accurate the results could be by applying the SMO. The rest of this paper is organised as follows. The details of the research method including datasets description, feature selection and training algorithm are covered in Section 2. Results and discussion are given in Section 3. This paper will be ended up with a conclusion.

2. RESEARCH METHOD

This section covers the details of the research method as depicted in Figure 1. By this method, a supervised learning is used to build the detection model produced to separate between email classes: spam and ham. In this work, the focus is on the textual content (body) of emails itself, i.e. the only information considered is contained in the body of the email being analysed.

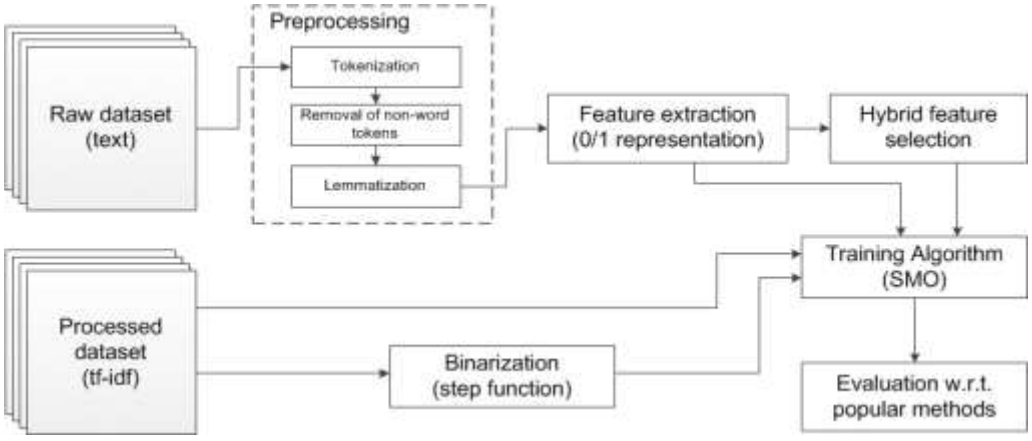

Figure 1. Sketch of the research method

2.1. Datasets description

In this paper, two datasets which have different representations of emails are used. The first one contains preprocessed data of about (960) emails in the form of text. This data set is balanced where spam and ham emails are equally distributed. Also, the emails are passed through preprocessing steps as follows:

a) *Tokenization*: The textual content of emails is split into words (tokens) for further considera-tion.
b) *Removal of non-token words*: Words which are common in English but have no meaning such as "the", "and" and "of" are removed, in addition numbers and punctuation are re-moved. These words have no impact on deciding whether the email being analysed is spam or ham. Also, white spaces such as tabs, newlines and spaces have all been trimmed to a single space.
c) *Lemmatization*: Words having the same meaning but different forms are adjusted to a single form (return them to their root). For example, the words "include", "includes" and "included" are replaced by "include". Additionally, all words in the body of emails are ensured to be in lower case form.

The second data set has an information about (4601) emails each of which is represented by a vector of (57) features in the form of term frequency-inverse document frequency (TF-IDF). The emails in this dataset is partitioned to (1813) spam emails and (2788) ham emails. Note that each email in both datasets is associated with a label denoting the class to which the email belongs.

2.2. Feature extraction

Feature extraction plays an important role in the process of classifying the textual content of documents such as emails. Through the feature extraction, we are looking for a representation which makes emails distinguishable. As previously mentioned the content of emails is processed in order to extract an important information which can be used to classify incoming emails into spam and ham. Since the content of the emails is unstructured data, a transformation (feature extraction) is applied to make it appropriate for further processing (detection). In this transformation, a vector representation (also called feature vector) is generated. Each entry in this vector corresponds to a feature in the email being transformed. Several methods have been proposed for feature extraction (for more details see [21, 22]).

In the present approach, a binary representation for the values of entries in the feature vector is used. Assume that B = $\{b_1, b_2, \ldots, b_k\}$ is a set of all different words that appears in all emails of the dataset. Then, given an email document e_i, a n-dimensional feature vector $\mathbf{x}_i = \{x_1, x_2, \ldots, x_k\}$ is generated. A feature x_i equals 1 if the corresponding word $b_i \in B$ appears in e_i, and 0 otherwise. In case of the processed dataset, obtaining a binary representation requires the application of a step function defined as follows: if the feature x_i is greater than 0 then, assign 1 to x_i, otherwise assign 0, see Figure 1.

2.3. Hybrid feature selection

To reduce the dimensionality and selecting the features which are relevant for the purpose of detection, a feature selection is applied. Through the process of selection, the most representative features are selected and then they will be used for predicting th e target class for a given email. Let the set *B* as defined above,

a new set $B' \subset B$ is produced such that the information about emails classes are still reserved. Various methods for feature selection have been developed [23-27]. In this paper, a hybrid method is developed using two of these methods, namely Gain Ratio and Correlation. A pseudocode of this developed method is given in Algorithm 1. Given a dataset *D* with *k* features and three threshold values, Algorithm 1 determines the best subset of features X_s with $|X_s| = k_1$, where $k_1 \leq k$. Staring from line 3, two sets of weights is assigned to both W^G and W^C. These sets of weights are obtained by applying both Gain Ratio and Correlation method. Then, the weights generated by these methods are used to test, against the three predefined thresholds Θ, Θ_1 and Θ_2, whether a certain feature is selected (added to a set of selected features X_s). All details about the testing process are included in the algorithm (line 6-9).

Algorithm 1 : Hybrid Feature Selection.
Input: $D = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$ is a dataset in which every $(\mathbf{x}_i, y_i) \epsilon\, D$ is such that $\mathbf{x}_i \epsilon \{0,1\}^k$, where *k* is the number of features. Θ, Θ_1 and Θ_2 are predefined thresholds.
Output: X_s is a set of selected features with $|X_s| = k_1$ and $k_1 \leq k$.

1: **Initialise** $X_s = \emptyset$
2: **Let** $W^G, W^C \in R^k$
3: $W^G = Gain\_Ratio(D)$
4: $W^C = Correlation(D)$
5: **for all** features $x_i, i \in \{1, \ldots, k\}$ **do**
6: **if** $w_i^G > \Theta$ **and** $w_i^C > \Theta$ **then**
7: $X_s = X_s \cup x_i$
8: **else if** $w_i^G > \Theta_1$ **or** $w_i^C > |\Theta_2|$ **then**
9: $X_s = X_s \cup x_i$
10: **end if**
11: **end for**

2.4. Learning algorithm

In this section, the details of the learning algorithm adopted in this paper for spam email detection are covered. The output of this algorithm is a hyperplane that classifies the dataset into two categories. For this end, this algorithm addresses a quadratic program (QP) problem formu-lated as follows. Let $\mathbf{x}_i$ be the input training vector i and $\mathbf{y}_i$ be its label for all i = 1,…, m. Then the goal of this QP problem is to find a solution for α_i and α_j (also called Lagrange multipliers) which optimise the following objective function:

$$\max_{\alpha} F(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \mathbf{y}_i \mathbf{y}_j K(\mathbf{x}_i, \mathbf{x}_j)\, \alpha_i \alpha_j$$
$$\text{s.t.} \quad 0 \le \alpha_i \le C \quad \forall i \tag{1}$$
$$\sum_{i=1}^{m} \boldsymbol{y}_i\, \alpha_i$$

where C is a constant and K($\mathbf{x}_i$, $\mathbf{x}_j$) represents the kernel function used to determine the similarity between the vectors $\mathbf{x}_i$ and $\mathbf{x}_j$. There are several examples of this function; in this paper the polynomial form is used [28]. Note that the relationship between the α_i and $\mathbf{x}_i$ is one-to-one. The solutions obtained for α's are used to determine the normal vector **w** and the threshold b explained in the following:

$$\mathbf{w} = \sum_{i=1}^{m} \mathbf{y}_i\, \alpha_i\, \mathbf{x}_i \tag{2}$$

$$b = \mathbf{w}.\mathbf{x}_k - \mathbf{y}_k\,, \quad \text{for some } \alpha_k > 0 \tag{3}$$

Algorithm 2 : SMO learning algorithm.

Input: $T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$, a set of training instances, where $x_i \in R^k$ and k is the number of features; $y_i \in \{+1, -1\}$ is the label of x_i. *Tolerance*, C and ϵ are predefined constants.

Output: the bias b and Lagrangian multipliers α's.

Initialise $\alpha_i = 0\ \forall x_i$ s.t. $(x_i, y_i) \in T, b = 0, Nochange = 0$
while $NoChange < max\_nochange$ **do**
  $change = 0$
  for all $i = 1, \ldots, m$ **do**
    $f(x_i) = \sum_{j=1}^{m} \alpha_j \times y_j < x, x_i > + b$
    if $(y_i \times E_i < Tolerance$ ***and*** $\alpha_i < C)$***or***$(y_i \times E_i > Tolerance$ ***and*** $\alpha_i > C)$ **then**
      Select j from [1,n] randomly
      $f(x_j) = \sum_{i=1}^{m} \alpha_i \times y_j < x, x_j > + b$
      $E_j = f(x_j) - y_j$
      $\alpha_1 = \alpha_j$
      $\alpha_2 = \alpha_j$
      if $(y_i \neq y_j)$ **then**
        $L = \max(0, \alpha_j - \alpha_i)$
        $H = \min(C, C + \alpha_j - \alpha_i)$
      else
        $L = \max(0, \alpha_j - \alpha_i + C)$
        $H = \min(C, \alpha_j - \alpha_i)$
      end if
      if $(L = H)$ **then** continue to the next i
      end if
      $\eta = -2kernal(x_i, x_j) - kernal(x_i, x_i) - kernal(x_j, x_j)$
      if $(\eta \geq 0)$ **then** continue to the next i
      end if
      $\alpha_j = \alpha_j - \frac{y_i(E_i - E_j)}{\eta}$
      if $(\alpha_j > H)$ **then** $\alpha_j = H$
      end if

27: **if** ($\alpha_j < L$) **then** $\alpha_j = L$
28: **end if**
29: **if** ($|\alpha_j - \alpha_2| < \epsilon$) **then** continue to the next *i*
30: **end if**
31: $\alpha_i = \alpha_i + y_j \times y_i(\alpha_2 - \alpha_j)$
32: $b_1 = b - E_i - y_i(\alpha_i - \alpha_1) < x_i, x_i > -y_j(\alpha_j - \alpha_2) < x_i, x_j >$
33: $b_2 = b - E_j - y_i(\alpha_i - \alpha_1) < x_i, x_j > -y_j(\alpha_j - \alpha_2) < x_j, x_j >$
34: **if** ($\alpha_i > 0$ ***and*** $\alpha_i < C$) **then** $b = b_1$
35: **else if** ($\alpha_j > 0$ ***and*** $\alpha_j < C$) **then** $b = b_2$
36: **else** $b = \frac{b_1 + b_2}{2}$
37: **end if**
38: **if** ($Changes = 0$) **then** $NoChange = NoChange + 1$
39: **else** $NoChange = 0$
40: **end if**
41: **end if**
42: **end for**
43: $Changes = Changes + 1$
44: **end while**

Once α's and *b* are determined, the output of the training algorithm is obtained using

$$u = \sum_{j=1}^{m} \mathbf{y}_j \, \alpha_j K\left(\mathbf{x}_i, \mathbf{y}_j\right) - b \tag{4}$$

where xi is the training input vector and xj is the stored training vector. Then, for unknown vector x, the class to which the email belongs can be determined by finding.

$$F(\mathbf{x}) = sign(\mathbf{w}.\mathbf{x} - b) \tag{5}$$

To solve the QP problem in (1), the SMO method is applied. This method consists of three components whose functions are to 1) find a solution for the two Lagrange multipliers, 2) use a heuristic method to choose which will be optimised, and 3) computing the threshold b. The pseudocode of the entire method is described in Algorithm 2.

Given a training set T and parameters Tolerance, C and ϵ, the SMO algorithm produces the bias b and the Lagrangian multipliers α's. This algorithm uses an iterative method to solve the QP problem formulated above. This problem is decomposed into a number of smaller sub-problems each of which is then solved analytically. Briefly, without loss of generality the SMO algorithm proceeds as follows: i) compute the second Lagrange multiplier $\alpha 2$ which does not satisfy the Karush–Kuhn–Tucker (KKT) conditions, ii) select the first Lagrange multiplier $\alpha 1$ and optimise both ($\alpha 1$, $\alpha 2$) and iii) loop over step (i) and (ii) until a stopping criterion is reached, i.e. no more Lagrangian multipliers violate (KKT) conditions. Choosing two Lagrangian multipliers to optimise yields the SMO an advantage over the existing methods. This advantage consists in the ability to handle large and complex datasets with less computational resources.

3. RESULTS AND ANALYSIS

In this section, the performance of the SMO algorithm is tested against four common methods: *Bayes Net*, *Naive Bayes*, *Logistic Function* and the *standard SVM*. In our settings, four metrics are used for measuring the performance of the different methods namely precision, recall, F-measure and accuracy. Moreover, three-fold cross validation is applied on the two datasets mentioned in Section 2.1. All results obtained by working on these datasets are organised into two scenarios. In case of the SMO for these scenarios, the datasets have been trained with C = 0.87 and ϵ = 0.001. Note that all experiments are carried out by taking features corresponding to the (2556) most frequent words in all emails in case of the raw dataset and ignore the last three features of the processed dataset.

In the first scenario, the learning algorithms being studied are applied on the original datasets and the results which have been obtained are shown in Tables 1 and 2. These results in both tables illustrate that the SMO outperforms the other algorithms with regards to all evaluation metrics. Another observation is that the values of these metrics in case of raw dataset is higher than in case of the processed dataset. One indication is that the binary representation of features is more appropriate to express emails than TF-IDF

representation. On the other hand, Table 3 highlights that transforming the TF-IDF representation to binary representation has a slight effect on the performance of the SMO.

The second scenario is arisen to study the impact of reducing the features and then measure the performance. By applying the hybrid feature selection method of Section 2.3. on the raw dataset, the number of features has been reduced from (2556) features to only (32) features, i.e. the reduction rate was about (98%). Although obtaining a significant reduction rate, the results of the comparable algorithms have increased, see Table 4. Furthermore, the results still give a preference to the SMO over other algorithms.

Generally speaking, we observe that use of the binary representation is preferred for all algorithms. In addition, comparing the results obtained by learning algorithms before and after applying feature selection indicates an improvement see Tables 1 and 4. Another observation points out that the SMO method yields a better performance than the SVM method in the different scenarios.

Table 1. Evaluation in case of raw dataset

Learning algorithm	Precision	Recall	F-measure	Accuracy (%)
Bayes Net	0.97	0.97	0.97	96.979
Naive Bayes	0.968	0.968	0.968	96.770
Logistic Function	0.965	0.965	0.965	96.458
SVM	0.97	0.97	0.97	96.979
SMO	0.972	0.971	0.971	97.083

Table 2. Evaluation in case of the processed dataset

Learning algorithm	Precision	Recall	F-measure	Accuracy (%)
Bayes Net	0.899	0.899	0.898	89.871
Naive Bayes	0.844	0.796	0.798	79.634
Logistic Function	0.844	0.796	0.798	79.634
SVM	0.9	0.9	0.899	89.958
SMO	0.93	0.93	0.929	92.979

Table 3. Evaluation in case of the processed dataset of binary representation

Learning algorithm	Precision	Recall	F-measure	Accuracy (%)
Bayes Net	0.885	0.885	0.884	88.502
Naive Bayes	0.874	0.865	0.861	86.481
Logistic Function	0.931	0.931	0.931	93.131
SVM	0.926	0.926	0.925	92.588
SMO	0.933	0.933	0.933	93.305

Table 4. Evaluation in case of the raw dataset after applying the hybrid feature selection

Learning algorithm	Precision	Recall	F-measure	Accuracy (%)
Bayes Net	0.972	0.972	0.972	97.187
Naive Bayes	0.968	0.968	0.968	96.770
Logistic Function	0.979	0.979	0.979	97.916
SVM	0.976	0.976	0.9764	97.604
SMO	0.981	0.981	0.981	98.125

4. CONCLUSION

Email is an important tool to exchange messages between users. Spammers can use this tool to mislead users by sending them spam emails which are not supposed to receive. Therefore, it is important here to distinguish between spam and ham. The mailbox could receive many spams which represent suspicious behavior against users. Thus, protecting these emails from malicious access has become a necessary task to keep users' email safe. In this paper, the problem of detecting spam emails is addressed by developing a detection approach based on hybrid feature selection and the SMO methods. Using these methods leads to producing a simplified model, i.e. less computational cost needed for spam detection. The performance experiments point out that the developed approach outperforms its counterparts according to the applied evaluation metrics in case of balancing and unbalancing datasets. In addition, these experiments proved that the binary representation of features improved the obtained results. One future direction can be explored by extending the present work to the case where there are multiple types of spam emails.

REFERENCES

[1] W. Ying, Y. Kai, and Z. Jianzhong, "Using dbscan clustering algorithm in spam identifying," *2nd International Conference on Education Technology and Computer*, vol. 1, pp. V1–398–V1–402, 2010.

[2] S. Thirumuruganathan, "A detailed introduction to k-nearest neighbor (kNN) algorithm," *Retrieved March*, vol. 20, pp. 2012, 2010.

[3] J. S. Whissell and C. L. A. Clarke, "Clustering for semi-supervised spam filtering," *Proceedings of the 8th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference*, pp. 125-134, 2011.

[4] S. Zhu, W. Dong, and W. Liu, "Hierarchical reinforcement learning based on knn classification algorithms," *Int J Hybrid Inf Technol*, vol. 8, no. 8, pp. 175-184, 2015.

[5] M. Sheikhalishahi, et al., "Fast and effective clustering of spam emails based on structural similarity," *International Symposium on Foundations and Practice of* Security, pp. 195-211, 2016.

[6] J. Wu, and T. Deng, "Research in anti-spam method based on bayesian filtering," *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, vol. 2, pp. 887-891, 2008.

[7] M. N. Marsono, M. W. El-Kharashi, and F. Gebali, "Binary lns-based naive bayes inference engine for spam control: noise analysis and fpga implementation," *IET Computers & Digital Techniques*, vol. 2, no. 1, pp. 56-62, 2008.

[8] B. Issac and W. J. Jap, "Implementing spam detection using bayesian and porter stemmer keyword stripping approaches," *TENCON 2009-2009 IEEE Region 10 Conference*, pp. 1-5, 2009.

[9] S. B. Rathod and T. M. Pattewar, "Content based spam detection in email using bayesian classifier," *International Conference on Communications and Signal Processing*, pp. 1257-1261, 2015.

[10] L. Kang, et al., "Using naive bayes method to classify text-based email," *9th International Symposium on Parallel Architectures, Algorithms and Programming*, 2018, pp. 94-98, 2018.

[11] N. F. Othman, and W. I. S. W. Din, "Youtube spam detection framework using naive bayes and logistic regression," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 3, pp. 1508-1517, 2019.

[12] H. Drucker, D. Wu, and V. N. Vapnik, "Support vector machines for spam categorization," *IEEE Transactions on Neural networks*, vol. 10, no. 5, pp. 1048-1054, 1999.

[13] D. Sculley and G. M. Wachman, "Relaxed online svms for spam filtering," *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 415-422, 2007.

[14] C. Tseng, and M. Chen, "Incremental svm model for spam detection on dynamic email social networks," *International Conference on Computational Science and Engineering*, vol. 4, pp. 128-135, 2009.

[15] G. Caruana, M. Li, and M. Qi, "A mapreduce based parallel SVM for large scale spam filtering," *Eighth International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 4, pp. 2659-2662, 2011.

[16] V. Vishagini, and A. K. Rajan, "An improved spam detection method with weighted support vector machine," *International Conference on Data Science and Engineering*, pp. 1-5, 2018.

[17] E. G. Dada, et al., "Machine learning for email spam filtering: review, approaches and open research problems," *Heliyon*, vol. 5, no. 6, 2019.

[18] J. Platt, "Fast training of support vector machines using sequential minimal optimization," *Advances in Kernel Methods-Support Vector Learning, Eds. MIT Press*, 1998.

[19] M. A. Hearst, et al., "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18-28, 1998.

[20] N. Cristianini, J. and Shawe-Taylor, "An introduction to support vector machines and other kernel-based learning methods," *Cambridge university press*, 2000.

[21] T. S. Guzella, and W. M. Caminhas, "A review of machine learning approaches to spam filtering," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10206-10222, 2009.

[22] A. Adeleke, et al., "A two-step feature selection method for quranic text classification," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 16, no. 2, pp. 730-736, 2019.

[23] L. Yu, and H. Liu, "Feature selection for high-dimensional data: A fast correlation-based filter solution," Proceedings of the 20th international conference on machine learning, pp. 856-863, 2003.

[24] A. G. Karegowda, A. Manjunath, and M. Jayaram, "Comparative study of attribute selection using gain ratio and correlation based feature selection," *International Journal of Information Technology and Knowledge Management*, vol. 2, no. 2, pp. 271-277, 2010.

[25] J. Tang, S. Alelyani, and H. Liu, "Feature selection for classification: A review," *Data classification: Algorithms and applications*, p. 37, 2014.

[26] J. Cai, J. Luo, S. Wang, and S. Yang, "Feature selection in machine learning: A new perspective," *Neurocomputing*, vol. 300, pp. 70-79, 2018.

[27] B. N. Kumar, M. S. B. Raju, and B. V. Vardhan, "A novel approach for selective feature mechanism for two-phase intrusion detection system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 1, pp. 101-112, 2019.

[28] S. Shalev-Shwartz and S. Ben-David, "Understanding machine learning: From theory to algorithms," *Cambridge university press*, 2014.

BIOGRAPHIES OF AUTHORS

Ahmed Al-Ajeli received the BSc and MSc degrees in Computer Science from the University of Babylon, Iraq, in 1999 and 2002, respectively. After completing his MSc, he worked as an assistant lecturer at the Department of Computer Science, the University of Babylon. In 2017, he received his PhD in Computer Science from the University of Birmingham, the UK. Currently, he holds a lecturer position at Software Department, University of Babylon. His current research interests include fault diagnosis/prognosis in discrete-event systems, machine learning and software development.

Raaid Alubady received his Ph.D. degrees in Information Technology from the Universiti Utara Malaysia, in 2017. He got a Bachelor's degree in Computer Sciences from University of Babylon-Iraq, a Higher Diploma in Data Security from Iraqi Commission for Computers and Informatics-Iraq, and a Master's degree in Information Technology from UUM- Malaysia. Alubady is a lecturer at the Network Information Department, College of Information Technology, University of Babylon- Iraq. He is a member of IEEE and actively involved in IEEE activities. In addition, he is a member of the Internet Society Malaysia Chapter; a member of the Iraqi Association for IT Specialists, Iraq; and a reviewer of several international academic journals and conferences. Currently attached to the InterNetWorks Research Laboratory (IRL). Raaid current area of research focuses on the Future Internet (ICN and NDN), Wireless Networking/ MANET, Internet of Things, Routing Protocol, and Performance Analysis.

Eman Al-Shamery received the BSc and MSc degrees in Computer Science from the University of Babylon, Iraq, in 1998 and 2001, respectively. After completing her MSc, she worked as an assistant lecturer at the Department of Computer Science, the University of Babylon. In 2013, she received her PhD in Computer Science from the University of Babylon. Currently, she holds a professor position at Software Department, University of Babylon. Her current research interests include artificial intelligence, bioinformatics, machine learning, neural networks, deep learning and data mining.