Received: Added at production | Revised: Added at production | Accepted: Added at production **This is the Pre-Published Version.**

DOI: xxx/xxxx

ARTICLE TYPE

An Empirical Study of Supervised Email Classification in Internet of Things: Practical Performance and Key Influencing Factors

Wenjuan Li[1,2] | Lishan Ke[3] | Weizhi Meng*[1,4] | Jinguang Han[5]

[1]Institute of Artificial Intelligence and Blockchain, Guangzhou University, Guangdong, China
[2]Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong, China
[3]School of Mathematics and Information Science, Guangzhou University, Guangdong, China
[4]Department of Applied Mathematics and Computer Science, Technical University of Denmark, Lyngby, Denmark
[5]Jiangsu Provincial Key Laboratory of E-Business, Nanjing University of Finance and Economics, Nanjing, China

Correspondence
*Weizhi Meng, Department of Applied Mathematics and Computer Science, Technical University of Denmark, Denmark. Email: weme@dtu.dk

A preliminary version appears in Proc. IEEE International Conference on Communications (ICC), IEEE, pp. 7438-7443, 2015 [18]

Summary

Internet of Things (IoT) is gradually adopted by many organizations to facilitate the information collection and sharing. In an organization, an IoT node usually can receive and send an email for event notification and reminder. However, unwanted and malicious emails are a big security challenge to IoT systems. For example, attackers may intrude a network by sending emails with phishing links. To mitigate this issue, email classification is an important solution with the aim of distinguishing legitimate and spam emails. Artificial intelligence especially machine learning is a major tool for helping detect malicious emails, but the performance might be fluctuant according to specific datasets. The previous research figured out that supervised learning could be acceptable in practice, and that practical evaluation and users' feedback are important. Motivated by these observations, we conduct an empirical study to validate the performance of common learning algorithms under three different environments for email classification. With over 900 users, our study results validate prior observations and indicate that LibSVM and SMO-SVM can achieve better performance than other selected algorithms.

KEYWORDS:
Email Classification, IoT Security, Supervised Learning, Spam Detection, Artificial Intelligence

1 | INTRODUCTION

Internet of Things (IoT) empowers the connected world by allowing the connection among various objects such as physical devices, sensors, controllers and intelligent computer processors. Gartner report predicted that up to 5.8 billion enterprise and automotive IoT endpoints would be deployed by 2020 [11]. More organizations started adopting IoT focusing on the business outcomes of the technology. A research report found that 71% of enterprises are gathering data for IoT initiatives [3].

As it is not always possible for managers to monitor the IoT data, emails are an effective and widely used solution for IoT systems to notify and remind users according to the pre-defined policies. However, unwanted or spam emails

are a big threat for IoT security, in which cyber attackers would try to send malicious emails and induce users to click phishing links[17,35]. A successful phishing attack can infer users' personal data and credentials to their accounts, devices and system[4]. If an attacker collects the information from your contact list or social media, they can spam people around you. It is found that the spam occupied up to 66.34% of the total email traffic in Q1 2014[15], while the average percentage of spam in global email traffic could still reach 50.18% in Q2 2020[36]. Hence there is a great need to develop email classification methods, through identifying and filtering unwanted emails for current IoT systems.

With the development of adversarial software, attackers can utilize robot applications to automatically generate hundreds of malicious emails every day with different contents and source addresses. This situation makes some detection methods like list-based detection ineffective in practice. To mitigate this issue, a desirable solution needs to involve artificial intelligence technique. In the literature, machine learning has been widely studied in email classification[13,44]. Different supervised learning algorithms are examined such as Decision Tree[40], Support Vector Machine[2], Naive Bayes[24], k-nearest neighbor[10] and deep leaning[23]. However, it is well known that the performance of machine learning algorithms would be unstable and fluctuant according to concrete datasets[26,27]. In addition, previous work[25] also found several issues of supervised learning: 1) the email domain is dynamic, and 2) a large amount of labeled training data is required.

Motivation. Some more complex algorithms or systems[9,14,48] are proposed attempting to enhance the performance of supervised email classification. However, previous study[18] figured out that there is a lack of empirical study to investigate the practical performance of supervised learning, as most existing studies usually adopt datasets for performance evaluation. One main limitation is that environmental elements may be ignored. Motivated by previous work[18], we try to validate the obtained results, and adopt the same research questions in this work: a) What is the performance of SML classifiers in real environments? and b) How about users' attitude regarding email classification in reality?

Contributions. To validate the results from previous work[18], we follow the same steps and conduct a new empirical study to explore the above questions and investigate the practical performance of supervised learning. Different from the previous work, we involve some new algorithms in our empirical study, such as Feed-Forward Neural Network, Bidirectional Long Short Term Memory Network (LSTM), and Sequential Minimal Optimization-SVM (an improved SVM). Our empirical study includes more than 900 users in three different environments. The contributions can be summarized as follows.

- To facilitate the comparison, we follow the similar settings according to the previous work[18], and conduct a new empirical study under three different environments: a research institute, a university and an IT company. We consider some popular supervised learning algorithms including Naive Bayes, Decision Tree, k-nearest neighbor (KNN) and Support Vector Machine (SVM), RBFNetwork, Feed-Forward Neural Network (FFNN), Bidirectional LSTM, and Sequential Minimal Optimization (SMO) with SVM.
- In our empirical study, SMO-SVM could outperform the other selected classifiers, by improving the training process for SVM classifier. Our study validates that environmental factors should be considered in designing email classification solutions. In addition, a practical evaluation and users' attitude should be considered when examining the classifier performance.

The rest of this paper is organized as follows. Section 2 introduces some related work regarding the application of machine learning in email classification. Section 3 describes our empirical study and discusses our results in different environments. In Section 4, we discuss the open challenges and future directions of supervised learning-based email classification, and we conclude our work in Section 5.

2 | RELATED WORK

Malicious emails like Phishing emails attempt to induce users to click on malicious links, which can provide intruders access to users' device, accounts, or personal information. By pretending to be a person or organization, spammers can easily infect users' device with malware or steal their credentials, e.g., passwords.

Email classification is an important and commonly used approach to detect and reduce unwanted / spam emails. The main idea is to distinguish the malicious emails from the legitimate ones. Generally, email classification can be categorized into *rule-based approaches* and *content-based approaches*.

Rule-based Approaches. This kind of classification can distinguish malicious / spam emails based on pre-defined rules. Its popularity is due to the simplicity, short processing time and no training data[12]. Two typical methods are based on whitelist and blacklist. The former refers to a list of accepted email addresses while the latter refers to a list of unaccepted email addresses. Sinha *et al.*[38] introduced four blacklists based on an academic environment, and presented that blacklists could show better false error rates than expected. Another detection model called *PreSTA* was proposed by West *et al.*[47], which combined the historical data (e.g., blacklist) and the spatial relationship of malicious IP addresses. Their results showed a 93% filtration rate on average. Moura *et al.*[30] provided an evaluation to explore the effect of third party BadHood blacklists. Liu *et al.*[22] introduced a detection system to identify spam based on involved telephone numbers. Their system combined unsupervised and supervised learning methods (with a small number of known spam numbers) to explore new spam numbers from a large dataset.

Although such kind of classification is efficient, the limitations are obvious: that is, such detection can be easily bypassed by spam variations due to the standard rules (or a static list). For example, attackers can forge an email's address and not be detected by a blacklist. Some other related work can refer to[8,33,34].

Content-based Approaches. This kind of classification can distinguish a malicious email based on distinctive content of emails. Thus, email classification can be considered as a task of binary text categorization, and various learning algorithms can be applied such as Naive Bayes, Decision Tree, k-nearest neighbor, Support Vector Machine (SVM) and more.

Naive Bayes. These classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong independence assumptions between the features. Marsono *et al.*[24] introduced a framework based on Naive Bayes classifier in the hardware level, using the Logarithmic Number System (LNS) to help reduce the computational burden. Another bayesian classification model was given by Chen *et al.*[6], including three modules: Aggregating One-Dependence Estimators (AODE), Hidden Naive Bayes (HNB) and Locally Weighted learning with Naive Bayes (LWNB). Their results indicated that the combined model, especially AODE and HNB, could perform well for spam filtration.

Decision Tree. This kind of classifier is commonly used in data mining, with the goal of predicting the value of a target variable based on several input variables. In particular, leaves represent class labels and branches represent conjunctions of features that lead to those class labels. Meizhen *et al.*[43] introduced a spam filtration based on fuzzy decision tree, which selects the behavioral features of emails using Information Gain. They extracted the features from email messages, and processed them by fuzzy processor and data generalization. Shi *et al.*[40] described an ensemble learning model based on decision tree. Their results on a public dataset presented a good classification rate. Zhang *et al.*[49] introduced a spam detection method aiming to reduce the false positive error. They used the wrapper-based feature selection method to extract crucial features, and used decision tree as the classifier model with C4.5 as the training algorithm. They later used the cost matrix to allocate different weights to two error types. Trivedi and Panigrahi[41] introduced a comparative study between decision tree classifiers, such as AD tree, decision stump and REP tree, with and without boosting algorithms like bagging, boosting with re-sample and AdaBoost.

K-Nearest Neighbor (KNN). This kind of classifier is a non-parametric method for classification and regression. It can locate the nearest neighbor in instance space and label the unknown instance with the same class label as that of the located (known) neighbor. For instance, Firte *et al.*[10] introduced a spam filter by using the KNN algorithm, which could provide a constant data and list update of most frequently words that appear in the messages. Prilepok and Kudelka[32] showed a nearest community classifier for spam detection by grouping similar emails, which could achieve 93.78% accuracy (with 80.72% of spam emails and 98.01% non-spam emails).

Support Vector Machines (SVMs). These are supervised learning models with associated learning algorithms, with the aim of analyzing data and recognizing patterns. The main idea is that, based on the labeled instances, the algorithm tries to find the optimal hyperplane, which can be used to classify new data points, i.e., it searches the most similar examples (support vectors) between classes.

Drucker *et al.*[7] studied the application of SVM in email classification based on binary features. As compared with Ripper, Rocchio, and boosting decision trees, their results on two datasets indicated that SVM could outperform the others. Sculley and Wachman[37] introduced an algorithm of Relaxed Online SVM (ROSVM), which could identify

TABLE 1 Basic information for each environment.

Environments	Participants	Occupation
Research Institute	143	Researchers and staff
Academic University	733	Students and teachers
Commercial Company	113	Personnel and managers

email spam and blog spam. Caruana *et al.*[5] provided a parallel SVM algorithm for scalable spam filtering, which could minimize the impact of accuracy degradation when distributing the training data amongst the SVM classifiers. Khamis *et al.*[16] analyzed the potentially email header features for spam detection, which classified the features using SVM. They found that SVM could reach an accuracy of 80%.

Discussions. In the current literature, many algorithms have been investigated regarding spam detection such as ensemble methods[42,50], semi-supervised learning[25,19] and hybrid multi-classifier[14]. These algorithms mostly can achieve better performance than the basic supervised learning classifiers. However, most of them were evaluated using datasets, which seldom contain environmental elements and could be evidently affected by the data size[42]. Also, previous work[18] observed that basic supervised classifiers like SVM could be accepted by most users in practice. In this work, our motivation is to validate such observations by performing a new empirical study.

3 | EMPIRICAL STUDY

In this section, similar to the previous work[18], we perform an empirical study with three different environments and make a comparison among several supervised classifiers.

3.1 | Methodology

To facilitate the comparison with previous observations, we study the performance of supervised learning under three environments (in Southern China): one research institute, one university and one IT company. To obey the privacy policy of each organization, Table 1 only briefly summarizes the general background of participants and environments.

Participants. The research institute has around 500 researchers and officers, and there are 7,000 students and 500 personnel in the university and company environment, respectively. Our study finally involves a total of 989 participants who are aged from 20 to 56. All the participants are volunteers and regular email users, with 40% females.

Similar to the previous work[18], we required the participants to use their official email accounts (with organizational domain) in the study, and our goals were explained before the start, including how we protect the data privacy.

Classifier Selection. The study goal is to investigate and validate the practical performance of supervised learning in email classification. For comparison with prior observations, we employ four commonly used supervised classifiers such as Naive Bayes, Decision Tree, k-nearest neighbor (KNN) and Support Vector Machines (SVM). To avoid implementation bias, these classifiers were extracted from the WEKA platform[46], which is a collection of machine learning algorithms. All classifiers adopted the default settings to explore the initial performance, and Table 2 shows the corresponding algorithms for each classifier category.

Different from the previous work[18], we also involved some new algorithms in our empirical study as follows.

- *Feed-Forward Neural Network (FFNN)*[31]. It consists of an input layer, two hidden layers, and an output layer. The information can go ahead in only one direction, forward, from the input nodes, through the hidden nodes (if any) and to the output nodes. There are no cycles or loops in the network.
- *Bidirectional LSTM (BiLSTM)*[23]. It is a kind of deep learning, which trains two LSTM models on the input sequence. The first one is remained original and the second one is a reversed copy of the input sequence. This can offer more context to the network, and lead to a faster and even fuller learning process to solve the classification problem.

TABLE 2 Specific algorithms for each category.

Category	Specific Algorithm
Naive Bayes	NaiveBayes
Decision Tree	J48
K-Nearest Neighbor	IBK
Neural Networks	RBFNetwork
Neural Networks	FFNN
Deep Learning	WekaDeeplearning4j
Neural Networks	FFNN
SMO-SVM	SMO-LibSVM

TABLE 3 Features extracted from users' emails.

Subject length	The number of receipts
Message size	The number of replies
The number of attachments	The level of importance
Type of attachments	The frequency of sending emails
Size of attachments	The frequency of receiving emails
The number of words in the subject	The name length of senders
The number of words in the message	The number of embedded images

- *Sequential Minimal Optimization (SMO) with SVM[1].* SMO can help optimize a minimal subset of just two points at each iteration, which is an algorithm for training SVM by solving the quadratic programming (QP) problem. At each step, SMO can select two elements to jointly optimize and find the optimal values for those two parameters given that all the others are fixed, and can update the values accordingly.

Feature Selection. To facilitate comparison, we adopted the same 14 features based on the previous effort [17,18] as shown in Table 3. In this study, we will compare the performance of classifiers based on these two feature sets.

Data Collection. In the empirical study, we used the same java tool from previous work [18], which can collect participants' emails from their accounts based on their user names and passwords. The tool is configured to collect all user emails within 6 months, and the empirical study was last from June 2018 - December 2019. The basic functions of this tool can be summarized as below:

- *Email collection.* The tool can collect all emails based on user accounts and passwords.
- *Feature extraction.* The tool can automatically extract features for all imported emails, which can be handled by a learning classifier.
- *Classification reports.* The tool can generate a report including all classification results.

For privacy protection, we did not perform the data collection directly, but sent the tool to all participants. All participants were introduced how to use the tool and provide us with results and feedback. We also ask all participants to check their emails and label spam emails based on the tool, so that a labeled and user-specific training dataset is available for each classifier. As users may have different views of spam emails, such data is very important to build an accurate classifier in practice.

Performance metrics. To measure the classification performance, this study uses several metrics such as AUC, FP, FN as below:

- *Area under an ROC curve (AUC).* This metric can represent the expected performance as a single scalar. Generally, the larger the AUC, the better the classification performance.

TABLE 4 Spam traffic in each environment.

Environments	Average Percentage of Spam
Research Institute	46.8%
Academic University	53.5%
Commercial Company	27.1%

- *False Positive Rate (FPR).* This metric indicates the ratio of how many legitimate emails are classified as a spam email.
- *False Negative Rate (FNR).* This metric indicates the ratio of how many spam emails are classified as legitimate.
- *Classification Accuracy (Acc).* This metric indicates the overall classification performance of both spam and legitimate emails.

Study Phases. Similar to the previous work, there are two phases in our empirical study:

- *Phase1.* To get initial classification performance for each classifier, the tool randomly selects 60% of emails for training and the remaining for testing with a 10-fold cross validation.
- *Phase2.* The tool utilizes all the labeled data to train each classifier and applies the classification model to preform email classification in corresponding environment for two weeks.

3.2 | Result Analysis

The classification performance is evaluated based on the collected results and users' feedback.

Spam Traffic. After the study, we collected the basic statistics from each participant. Table 4 shows the average percentage of spam traffic in each different environment. It validates the observation that the spam rate is different in distinct environments. For example, the spam rate is 46.8%, 53.5% and 27.1% for each environment, respectively. Similar to prior observations[18], below are some major factors that may affect the spam rate in different conditions.

- *Control policy.* Intuitively, network managers are responsible for defining or adopting policies to control traffic in each environment. For example, each environment has an IT department or a similar group that can help complete this task. Based on the spam rate, it is easily found that the control policy in a company is more strict than other environments, i.e., they used an email filter to reduce spams.
- *Subscription services.* After an informal interview with some participants, it is observed that subscription services may affect the spam rate. For example, students may use their university email to register some services and websites, which would leak their email addresses. Attackers can easily grasp a list of university email addresses from the Internet.
- *Public webpage.* For the research institute, it is found that most researchers and staff have their personal website or an introduction page on their institutional website. Attackers can often grasp these email addresses by using some automate tools. This increases the spam rate for the research institute environment.

Phase1 Result. The purpose of this phase is to study the performance of supervised learning in detecting malicious emails under different environments. Figure 1 and Figure 2 depict the classification performance, including false positive rate, false negative rate, AUC and accuracy.

For simplicity, we assign a single number to each environment, e.g., FN(1) refers to the false negative rate under the research institute, FN(2) refers to the false negative rate under the university environment, and FN(3) refers to the false negative rate under the company environment.

The results validate that the performance of supervised learning varied with concrete environments. Figure 1 shows that under the research institute, LibSVM and SMO-LibSVM could reach lower false rates than others (around 0.09

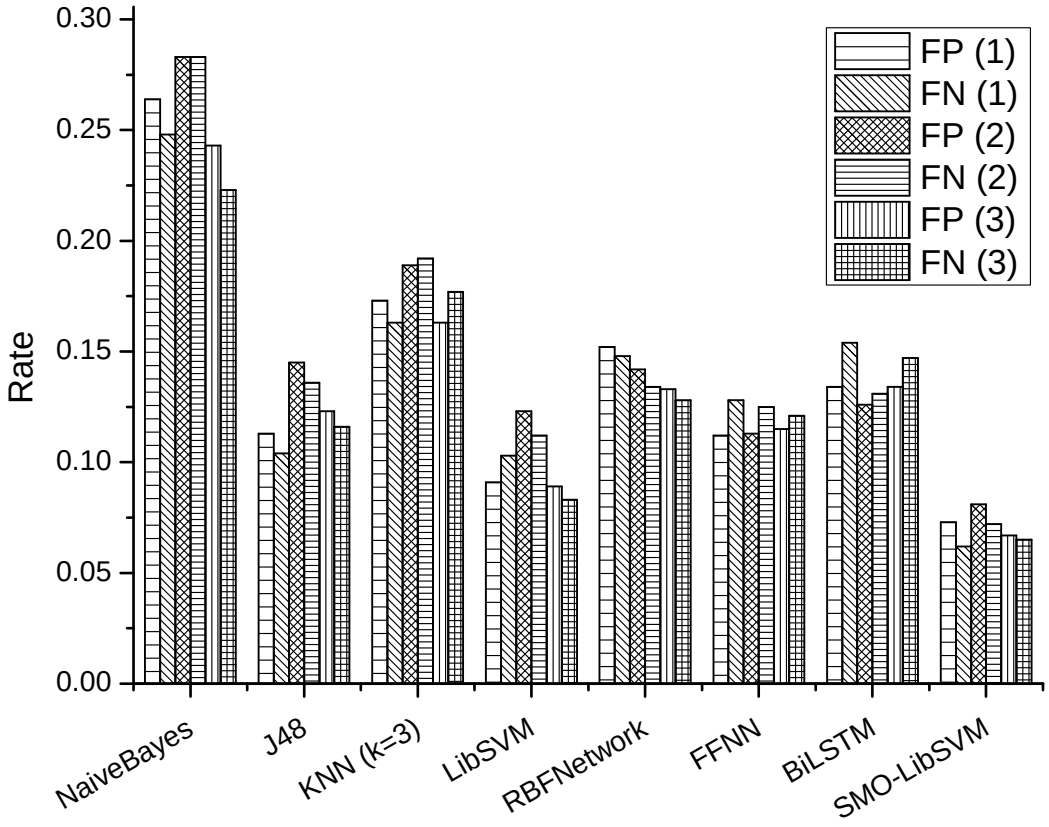

FIGURE 1 Phase1: classification performance with false positive rate and false negative rate.

for LibSVM and 0.07 for SMO-LibSVM). For other algorithms, J48 and FFNN could achieve a similar false rate of 0.11. For neural networks, FFNN could outperform RBFNetwork (0.11 vs. 0.15). The results are similar to the university environment and the company environment.

Figure 2 shows the AUC and accuracy of all algorithms, which are inline with the observations in Figure 1. SMO-LibSVM could achieve the best result, i.e., 0.92 under research institute, 0.91 under the university environment and 0.94 under the company environment. This is because SMO can optimize the training process for SVM. The original SVM could provide an accuracy rate of 0.90, 0.88 and 0.91. In addition, both J48 and FFNN could reach an accuracy rate above 0.88.

Similar to the previous work[18], a low classification rate was found under the university environment. Below are three main reasons:

- *Usage control.* The control policy is relatively flexible under the university environment, and students often use their email address in many places like a forum. In addition, the filtration policy in the university is not that strict as compared with the institute and company environment.
- *Complexity and diversity.* Due to the usage flexibility, the incoming emails are more complex in the university environment. In other words, attackers can get a university email address more easily. With a wider exposure of email address, the incoming emails will become more diverse, i.e., from a wider range of senders.
- *Classifier training.* The complexity and diversity of incoming emails can increase the difficulty of building an accurate classification model in the university environment. That is, incoming emails can have more variations of domains and contents, which may degrade the classification performance.

Phase2 Result. This phase attempts to investigate the classification performance with a retention period. That is, we used all the collected data as training data and built a model for each classifier. Then, we used the established

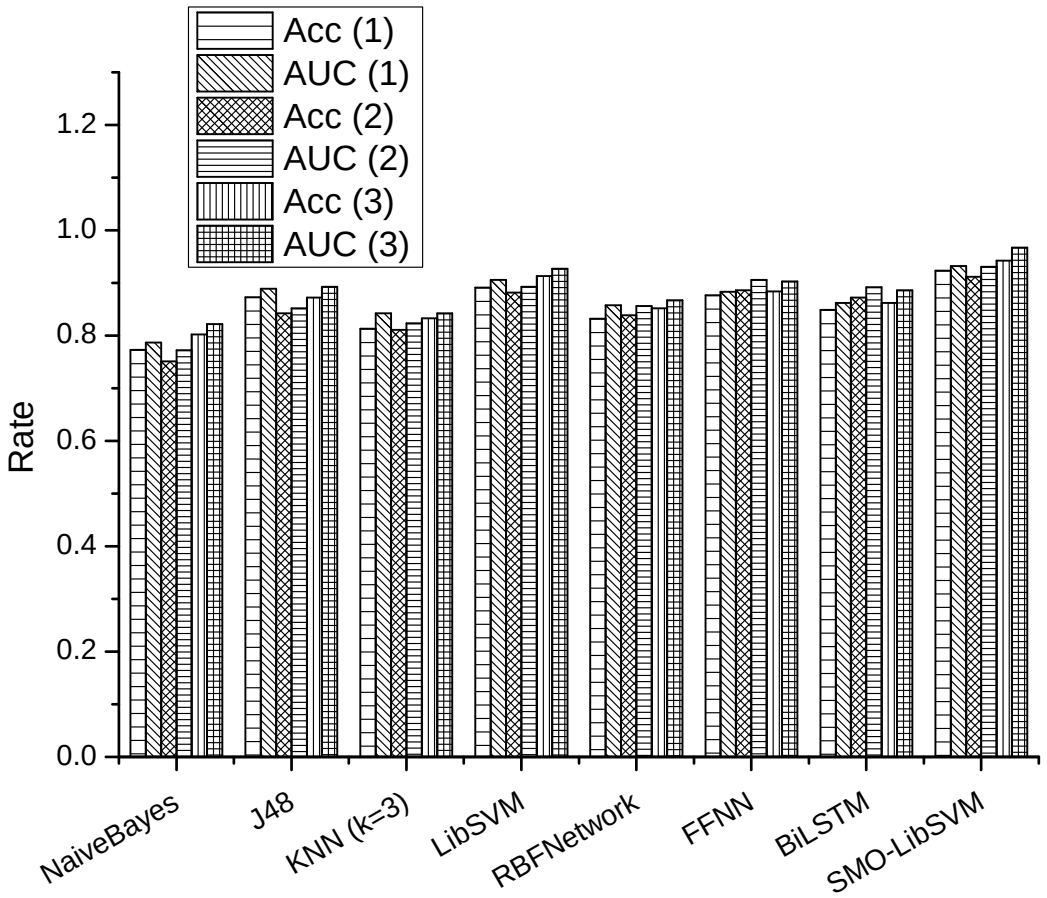

FIGURE 2 Phase1: classification performance with AUC and accuracy.

model to classify emails in the next two weeks. Figure 3 and Figure 4 present the classification performance of each classifier under different environments.

As compared with the results in Figure 1 and Figure 2, it is found that classifiers could perform better in this phase. This indicates that the increase of training data can enhance the classification performance. Similarly, SMO-LibSVM could achieve the best performance among these classifiers, i.e., it achieved an accuracy rate of 0.932, 0.935 and 0.957 for respect environment. While the performance of J48 and FFNN was also enhanced, i.e., J48 provided an accuracy rate of 0.883, 0.862 and 0.892; FFNN provided an accuracy rate of 0.885, 0.892, and 0.898. We notice that the accuracy rate of BiLSTM is 0.86, 0.84 and 0.88 respectively, which needs further parameter optimization.

In addition, all classifiers could reach the best performance under the company environment, but suffer the worst rate under the university environment. This is because the email complexity and diversity is low in the company, but high in the university scenario.

3.3 | User Feedback

Similar to previous work[18], we advocate that users' feedback is one important factor to evaluate the performance of supervised classifiers. In the empirical study, we provided each participant with some Likert-scale questions, regarding their opinions on the usage of supervised classifiers. In particularly, 1-score indicates strong dissatisfaction and 10-score indicates strong satisfaction. Table 5 summarizes the feedback and average scores for each classifier.

It is found that SMO-LibSVM received the best score of 8.9 than the other algorithms. After an informal survey, most participants gave a positive response regarding the usage of SMO-LibSVM in the aspects of accuracy and time consumption. The original LibSVM received many positive responses as well, with the second highest score of 8.2. Due to the unstable performance, J48, FFNN and BiLSTM obtained a score of 7.9, 7.5 and 7.2. With a low score,

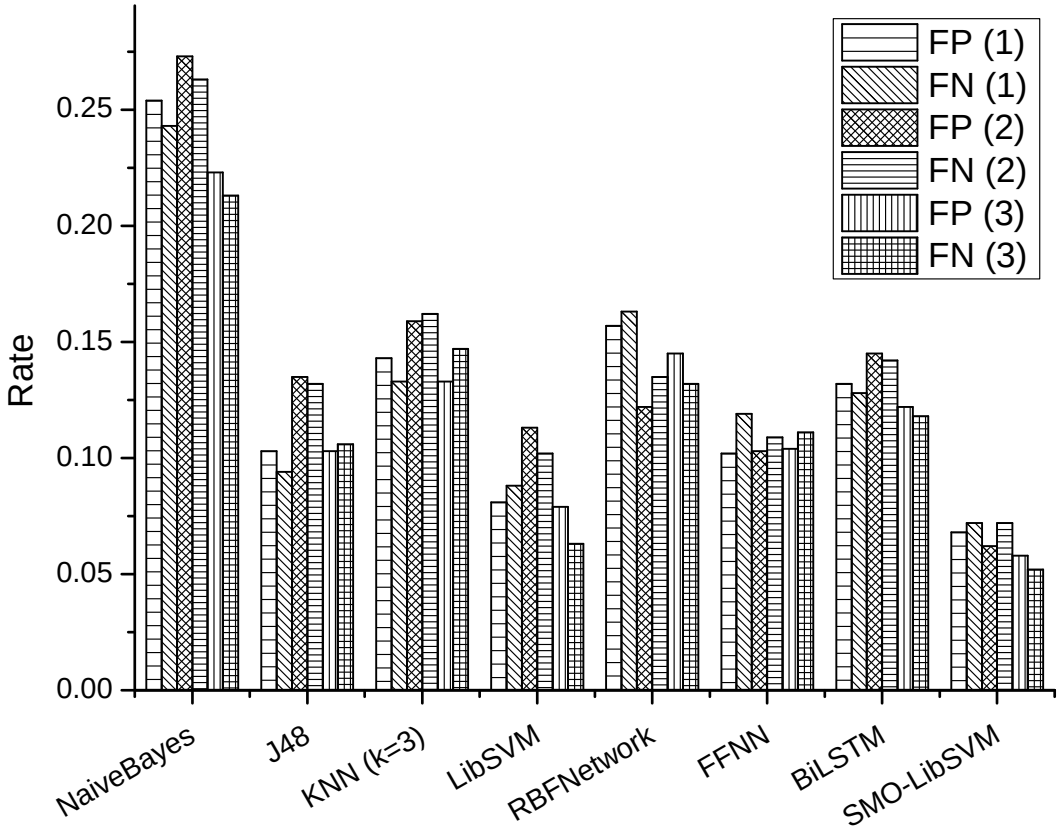

FIGURE 3 Phase2: classification performance with false positive rate and false negative rate.

TABLE 5 Users' feedback and average score for each classifier.

Supervised Classifiers	Average Score
NaiveBayes	2.6 ± 0.4
J48	7.9 ± 1.2
IBK (3)	3.1 ± 1.2
LibSVM	8.2 ± 1.1
RBFNetwork	6.9 ± 1.5
FFNN	7.5 ± 1.4
BiLSTM	7.2 ± 1.5
SMO-LibSVM	8.9 ± 0.5

participants considered that NaiveBayes and IBK may not work well in practice, at least these algorithms need to be greatly optimized.

Regarding the deviation, SMO-LibSVM and NaiveBayes have the lowest value, as compared with other classifiers. Table 6 further analyzes users' feedback regarding LibSVM and SMO-LibSVM under different environments. It is observed that participants in the company environment often gave a higher score than those in other environments. This is because both classifiers could perform well in the company environment.

The deviation of LibSVM is higher than SMO-LibSVM. There are two main reasons: 1) LibSVM could achieve good classification for most participant's emails but not all, i.e., due to the complexity and diversity of incoming emails per user. 2) SMO-LibSVM uses SMO algorithm to optimize the training process for SVM in order to reduce

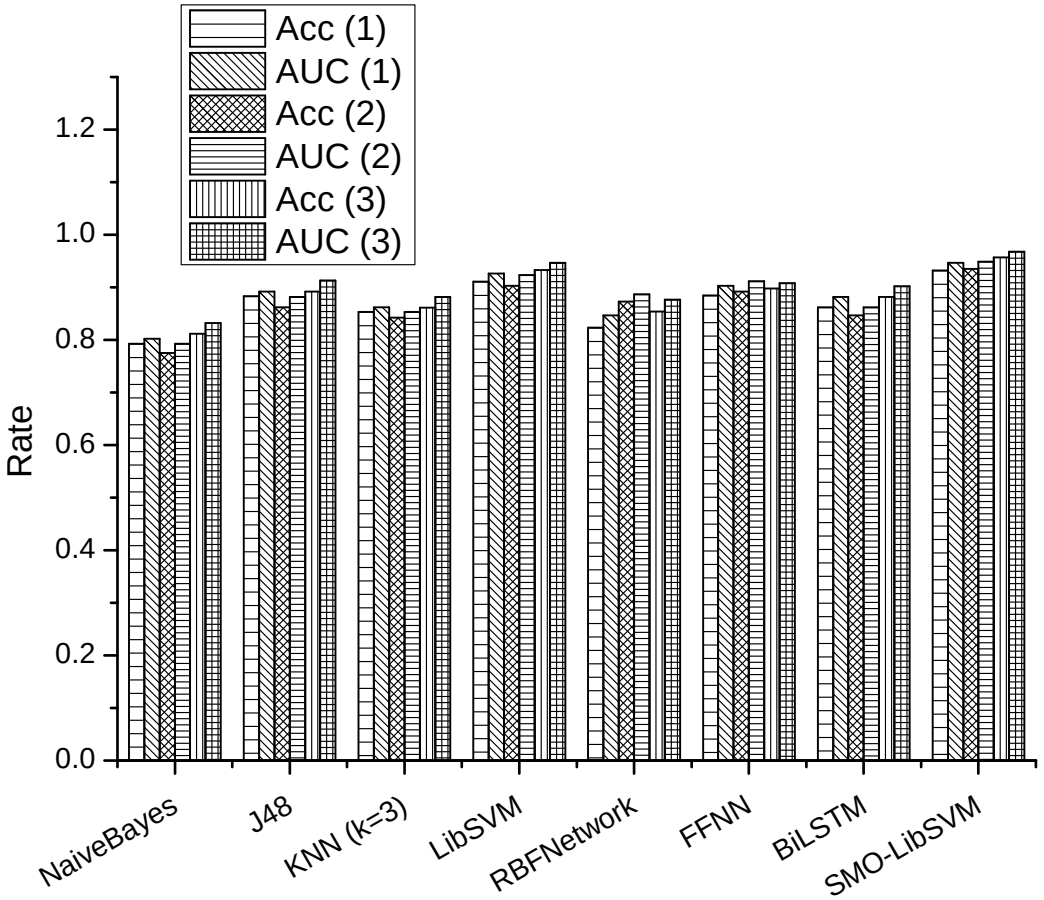

FIGURE 4 Phase2: classification performance with AUC and accuracy.

TABLE 6 Users' feedback for different environments.

Environments/Scores	LibSVM	SMO-LibSVM
Research Institute	7.9 ± 0.9	8.9 ± 0.3
Academic University	8.1 ± 1.3	8.7 ± 0.5
IT Company	8.4 ± 1.1	9.0 ± 0.4

the loss, which can build a more stable model and be adaptive to the environment. Overall, our collected feedback shows that LibSVM and SMO-LibSVM are accepted by most participants in practice.

As compared with the results in previous study[18], it is a bit surprised that J48 did not work well this time, i.e., it received 8.3 in former work[18] versus 7.9 in our empirical study. This highlights our motivation that classifier performance needs to be examined in practical environments, and should consider users' feedback.

4 | DISCUSSION

4.1 | Observations

In this empirical study, we investigate the performance of several supervised learning classifiers under three environments, such as research institute, university and IT company. Our results validate some observations in previous study[18] as follows. 1) Classifier performance would be unstable in different environments. 2) The incoming emails

are more complex in the university environment, and the classifier performance is worse than that in other environments. 3) Most classifiers could reach a better classification rate in the company environment, due to some additional security mechanisms adopted by the company. 4) Traditional supervised algorithms like LibSVM could be accepted by most users in a real environment.

Differently, it is found that the optimized algorithm of SMO-LibSVM could outperform other classifiers, as it can use SMO algorithm to improve the training efficiency and minimize the informaiton loss. In addition, J48 only received a low score in comparison with the score obtained in the prior study[18]. This reflects that environmental factor needs to be considered in designing an email classification solution.

Also, we validate that users' feedback is an important factor to help evaluate the performance and usability of a classifier. It is noticed that many more complicated algorithms are being developed for email classification, while the usability (or empirical study) should be considered in the evaluation. Further, we notice that users may have their own attitude and opinions to judge a classifier, in which the feedback might be different on the same classifier with different users.

4.2 | Open Challenges

Our empirical study highlights the importance of environmental factors when designing an email classification method. In addition, we also identify some open challenges in this field.

Complexity and diversity. Different network environments may have special spam rate according to the complexity and diversity of incoming emails. For example, our study shows that the university environment often has a higher spam rate with more complexity and diversity, than a research institute and a company. Under such environment, it is hard to build an accurate model for email classification. How to reduce the complexity and diversity is an open challenge.

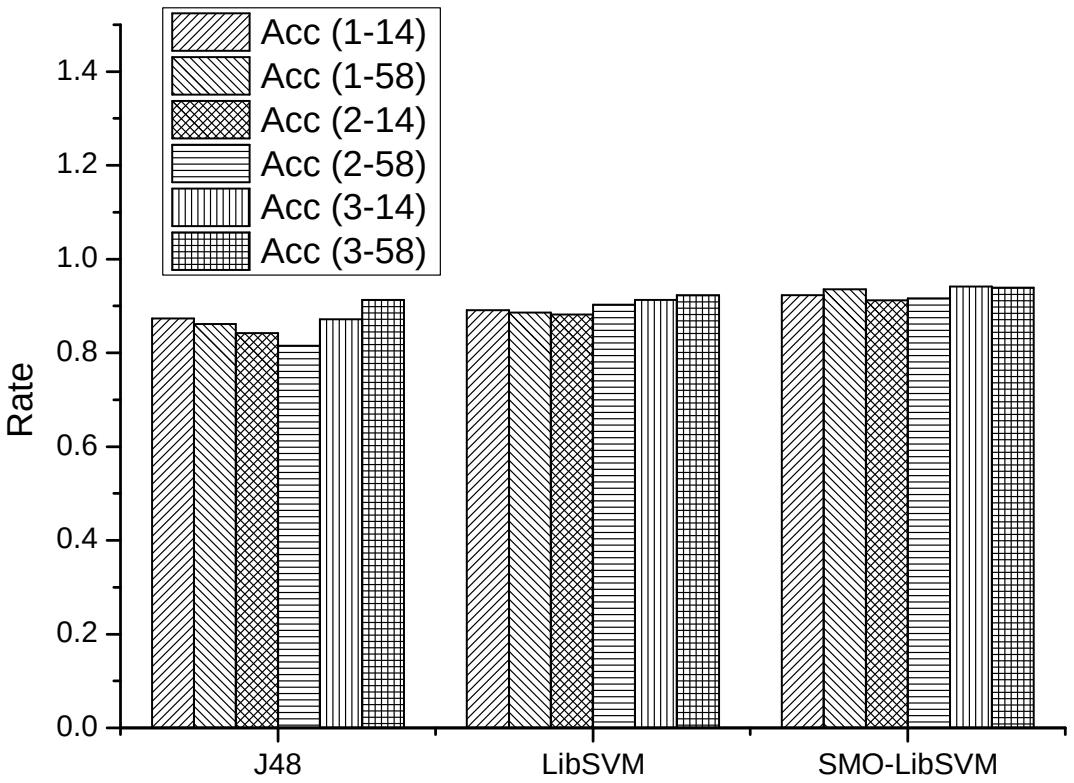

FIGURE 5 A comparison among two different feature sets.

Feature selection. Different classification methods may use a special set of features extracted from an email. For example, *SPAM E-mail Dataset*, a publicly available spam email dataset [39] consists of 58 features. While it is unclear whether all these features can contribute to the classification.

As J48, LibSVM and SMO-LibSVM received better feedback than other algorithms in our study, we further compare their performance based on two feature sets: 14 features (in our study) and 58 features (in the dataset). Our developed tool extracts these features based on the imported emails and tested on the data in *Phase1*. Figure 5 depicts the performance comparison between these two feature sets. It is found that the classifier performance would not be always enhanced with more features. For example, in the company environment, J48 could be enhanced with 58 features, but its performance with 58 features was degraded under the other environments. The performance of LibSVM with more features could be enhanced in the university environment and the company environment, while more features are beneficial for SMO-LibSVM in the research institute and the university environment.

4.3 | Potential Enhancement

Below are some potential directions for enhancing the supervised email classification.

- *Collaborative intrusion detection.* To complement current email filtration techniques, collaborative intrusion detection is a basic and essential security mechanism for IoT systems, which can allow information and data exchanged among different nodes [20,21,29].
- *Environment-focused scheme.* Our study has validated that environmental factor can affect the classifier performance in practice. Thus there is a need to consider such factor in future scheme design. For instance, we need to consider a specific scenario when designing an email classification scheme.
- *Adaptive algorithm design.* In the literature, it is recognized that the classifier performance would not be stable under different scenarios. Thus, we need to consider some intelligent or adaptive algorithm / mechanism [28,45] that can maintain the classifier performance.
- *Practical evaluation.* Our study validated that practical evaluation (in a real environment) is very essential for a classifier, as the real email traffic would be more complicated than a dataset, especially under some environments.
- *User feedback.* Our study figured out that users' feedback (in an empirical study) is an important factor to determine whether an email classification method is suitable in practice. Hence such factor should be considered in our future study.

5 | CONCLUSION

Emails are often used in an IoT system to notify or remind people of special events. However, with the rapid increase of IoT nodes, unwanted or spam emails have become a big security threat to IoT users. Thus, email classification is an important solution to refine incoming emails, and machine learning techniques are widely adopted in classifying spam emails, but the performance would be not stable depending on particular data items. Motivated by previous work, this work aims to conduct a new empirical study with over 900 users and investigate the performance of several learning algorithms under three different environments, including Naive Bayes, Decision Tree, k-nearest neighbor (KNN) and Support Vector Machine (SVM), RBFNetwork, FFNN, BiLSTM, and SMO-SVM. Our results demonstrated that SMO-SVM could outperform other classifiers and the environmental factor is important for designing an email classification method. In addition, users' feedback should be considered to examine the performance of a classifier.

ACKNOWLEDGEMENTS

This work was partially supported by National Natural Science Foundation of China (No. 61802077).

References

1. J.H. Abawajy, A.V. Kelarev: A Multi-tier Ensemble Construction of Classifiers for Phishing Email Detection and Filtering. CSS 2012, pp. 48-56, 2012.
2. O. Amayri and N. Bouguila, "A study of spam filtering using support vector machines," *Artificial Intelligence Review* 34(1), pp. 73-108, 2010.
3. 451 Research (access on September 2020) https://www.iot-now.com/tag/451-research/
4. J. Brutlag and C. Meek, "Challenges of the email domain for text classification," *Proceedings of ICML*, pp. 103-110, 2000.
5. G. Caruana, M. Li, M. Qi: A MapReduce based parallel SVM for large scale spam filtering. FSKD 2011: 2659-2662
6. C. Chen, Y. Tian, and C. Zhang, "Spam filtering with several novel bayesian classifiers," *Proceedings of ICPR*, pp. 1-4, 2008.
7. H. Drucker, D. Wu, and V.N. Vapnik, "Support vector machines for spam categorization," *IEEE Transactions on Neural Networks* 10(5), pp. 1048-1054, 1999.
8. Z. Duan, Y. Dong, and K. Gopalan, "DMTP: Controlling spam through message delivery differentiation," *Computer Networks* 51(10), pp. 2616-2630, 2007.
9. O.M.E. Ebadati, F. Ahmadzadeh: Classification Spam Email with Elimination of Unsuitable Features with Hybrid of GA-Naive Bayes. J. Inf. Knowl. Manag. 18(1): 1950008 (2019)
10. L. Firte, C. Lemnaru, and R. Potolea, "Spam detection filter using KNN algorithm and resampling," *Proceedings of ICCP*, pp. 27-33, 2010.
11. Gartner report (access on September 2020) https://www.gartner.com/en/newsroom/press-releases/2019-08-29-gartner-says-5-8-billion-enterprise-and-automotive-io
12. R. Hunt and J. Carpinter, "Current and New Developments in Spam Filtering," *Proceedings of ICON*, pp. 1-6, 2006.
13. R. Islam and Y. Xiang, "Email Classification Using Data Reduction Method," *Proceedings of ChinaCom*, pp. 1-5, 2010.
14. M.R. Islam, W. Zhou, M. Guo, and Y. Xiang, "An innovative analyser for multi-classifier e-mail classification based on grey list analysis," *Journal of Network and Computer Applications* 32, pp. 357-366, 2009.
15. "Spam in Q1 2014: US Once Again the Prime Target for Malicious Emails," Online, May, 2014. Available: http://www.kaspersky.com/about/news/spam/2014/Spam-in-Q1-2014-US-Once-Again-the-Prime-Target-for-Malicious-Emails.
16. S.A. Khamis, C.F.M. Foozy, M.F.A. Aziz, N. Rahim: Header Based Email Spam Detection Framework Using Support Vector Machine (SVM) Technique. SCDM 2020: 57-65
17. W. Li, W. Meng, Z. Tan, and Y. Xiang, "Towards Designing An Email Classification System Using Multi-View Based Semi-Supervised Learning," *Proceedings of TrustCom*, pp. 174-181, 2014.
18. W. Li and W. Meng. An Empirical Study on Email Classification Using Supervised Machine Learning in Real Environments. The 2015 IEEE International Conference on Communications (ICC 2015), IEEE, pp. 7438-7443, 2015.
19. W. Li, W. Meng, Z. Tan, and Y. Xiang, "Design of Multi-View Based Email Classification for IoT Systems via Semi-Supervised Learning," Journal of Network and Computer Applications, vol. 128, pp. 56-63, 2019.

20. W. Li, S. Tug, W. Meng, and Y. Wang. Designing Collaborative Blockchained Signature-based Intrusion Detection in IoT environments. Future Generation Computer Systems, vol. 96, pp. 481-489, 2019.

21. W. Li, W. Meng, and M.H. Au. Enhancing Collaborative Intrusion Detection via Disagreement-based Semi-Supervised Learning in IoT environments. Journal of Network and Computer Applications, vol. 161, 102631, pp. 1-9, Elsevier, 2020.

22. J. Liu, B. Rahbarinia, R. Perdisci, H. Du, L. Su: Augmenting Telephone Spam Blacklists by Mining Large CDR Datasets. AsiaCCS 2018: 273-284

23. S. Liu, K. Lee, I. Lee: Document-level multi-topic sentiment classification of Email data with BiLSTM and data augmentation. Knowl. Based Syst. 197: 105918 (2020)

24. M.N. Marsono, M.W. El-Kharashi, and F. Gebali, "Binary LNS-based naive Bayes hardware classifier for spam control," *Proceedings of IEEE International Symposium on Circuits and Systems*, pp. 3674-3677, 2006.

25. Y. Meng, W. Li, and L.F. Kwok, "Enhancing Email Classification Using Data Reduction and Disagreement-based Semi-Supervised Learning," *Proceedings of ICC*, pp. 622-627, 2014.

26. Y. Meng, L.F. Kwok. Enhancing False Alarm Reduction Using Voted Ensemble Selection in Intrusion Detection. International Journal of Computational Intelligence Systems, vol. 6, no. 4, pp. 626-638, Taylor & Francis, May 2013.

27. Y. Meng, L.F. Kwok. Adaptive Non-Critical Alarm Reduction Using Hash-based Contextual Signatures in Intrusion Detection. Computer Communications, vol. 38, pp. 50-59, Elsevier, 2014.

28. Y. Meng and L.F. Kwok. Adaptive Blacklist-based Packet Filter with A Statistic-based Approach in Network Intrusion Detection. Journal of Network and Computer Applications, vol. 39, pp. 83-92, 2014.

29. W. Meng, W. Li and L.F. Kwok, EFM: Enhancing the Performance of Signature-based Network Intrusion Detection Systems Using Enhanced Filter Mechanism. Computers & Security, vol. 43, pp. 189-204, 2014.

30. G.C.M. Moura, A. Sperotto, R. Sadre, and A. Pras, "Evaluating third-party Bad Neighborhood blacklists for Spam detection," *Proceedings of IFIP/IEEE IM*, pp. 252-259, 2013.

31. M. Paliwal, U.A. Kumar: The predictive accuracy of feed forward neural networks and multiple regression in the case of heteroscedastic data. Appl. Soft Comput. 11(4), pp. 3859-3869, 2011

32. M. Prilepok, M. Kudelka: Spam Detection Based on Nearest Community Classifier. INCoS 2015: 354-359

33. A. Ramachandran, N. Feamster, and S. Vempala, "Filtering spam with behavioral blacklisting," *Proceedings of ACM CCS*, pp. 342-351, 2007.

34. Z. Sadan and D.G. Schwartz, "Social network analysis of web links to eliminate false positives in collaborative anti-spam systems," *Journal of Network and Computer Applications* 34(5), pp. 1717-1723, 2011.

35. T. Shcherbakova and M. Vergelis, "Spam report: February 2014," Online, March 2014. Available at: http://securelist.com/analysis/monthly-spam-reports/58559/spam-report-february-2014/.

36. Spam and phishing in Q2 2020. https://securelist.com/spam-and-phishing-in-q2-2020/97987/

37. D. Sculley and G.M. Wachman, "Relaxed Online SVMs for Spam Filtering," *Proceedings of ACM SIGIR*, pp. 415-422, 2007.

38. S. Sinha, M. Bailey, and F. Jahanian, "Shades of Grey: On the effectiveness of reputation-based blacklists," *Proceedings of MALWARE*, pp. 57-64, 2008.

39. Spam dataset. http://web.cs.wpi.edu/~cs4445/b12/Datasets/spambase.arff.

40. L. Shi, Q. Wang, X. Ma, M. Weng, and H. Qiao, "Spam email classification using decision tree ensemble," *Journal of Computational Information Systems* 8(3), pp. 949-956, 2012.

41. S.K. Trivedi, P.K. Panigrahi: Spam classification: a comparative analysis of different boosted decision tree approaches. J. Syst. Inf. Technol. 20(3): 298-105 (2018)

42. J. Wang, K. Gao, Y. Jiao, and G. Li, "Study on ensemble classification methods towards spam filtering," *Proceedings of ADMA*, pp. 314-325, 2009.

43. M. Wang, Z. Li, S. Zhong: A Method for Spam Behavior Recognition Based on Fuzzy Decision Tree. CIT (2) 2009: 236-241

44. S. Wasi, S.I. Jami, Z.A. Shaikh: Context-based email classification model. Expert Syst. J. Knowl. Eng. 33(2): 129-144 (2016)

45. Y. Wang, W. Meng, W. Li, Z. Liu, Y. Liu, and H. Xue. Adaptive Machine Learning-based Alarm Reduction via Edge Computing for Distributed Intrusion Detection Systems. Concurrency and Computation: Practice and Experience, vol. 31, no. 19, Wiley, 2019.

46. The University of Waikato. WEKA-Waikato Environment for Knowledge Analysis. http://www.cs.waikato.ac.nz/ml/weka/

47. A.G. West, A.J. Aviv, J. Chang, and I. Lee, "Spam mitigation using spatio-temporal reputations from blacklist history," *Proceedings of ACSAC*, pp. 161-170, 2010.

48. W. Zhang, D. Zhu, Y. Zhang, G. Zhou, and B. Xu, "Harmonic functions based semi-supervised learning for web spam detection," *Proceedings of SAC*, pp. 74-75, 2011.

49. Y. Zhang, S. Wang, P. Phillips, G. Ji: Binary PSO with mutation operator for feature selection using decision tree applied to spam detection. Knowl. Based Syst. 64: 22-31 (2014)

50. Y. Zhen, N. Xiangfei, X. Weiran, and G. Jun, "An approach to spam detection by Naive bayes ensemble based on decision induction," *Proceedings of ISDA*, pp. 861-866, 2006.