Spam Mail Scanning Using Machine Learning Algorithm

Asma Bibi¹, Rasia Latif², Samina Khalid¹*, Waqas Ahmed², Raja Ahtsham Shabir¹, Tehmina Shahryar³

¹Department of Computer Science & IT, Mirpur University of Science and Technology, Pakistan.
²Department of Computer Science & IT, University of Kotli, Kotli Azad Kashmir, Pakistan.
³Department of Software Engineering, Mirpur University of Science and Technology, Pakistan.

*Corresponding author. Tel.: +923335859384; email: samina.csit@must.edu.pk
Manuscript submitted November 26, 2019; accepted January 24, 2020.
doi: 10.17706/jcp.15.2.73-84

Abstract: Emails are used in professional and personal level as a way of communication. With the passage of time emails are used for advertisement, spreading virus and fraud email for plaguing users of the internet. These type of unsolicited emails are categorized as spam and other legitimated emails are categorized as ham. Over the year several machine learning algorithms are used to predict emails category. In this paper we reflect on the classifier which is good for text classification. We evaluate machine learning algorithm on spam emails detection and outcomes shows naïve Bayes algorithm gives effective accuracy and precision using WEKA and our email management system which utilize php-ml library hosted by GitHub. Also comparative study with SVM and previous existing system in terms of accuracy and dataset used.

Key words: Email spam classification, Naïve Bayes, Support Vector Machine (SVM), spam filter, email classification.

1. Introduction

From the 1990's net started out to advantage reputation in world. In conjunction with the increase of the internet and e-mail, there was a dramatic growth in spam in the last years [1]. It became speedy diagnosed as a high-quality advertising tool at negligible cost. A person can use email system to send an e-mail message to thousands of people. regardless of social media, emails are still the essential source of private and professional communication. If email messages comprise an undesirable commercial, malicious code, phishing content, mass mailed and junk stuff, it's far generally known as spam or junk mail. Most dangerous attack is phishing attack that's cyberattack [2]. Through phishing attack hackers extract facts from the consumer which may be economic, personal or might be professional. The legitimated emails are referred to as ham, which are useful for the consumer and did not damage the user. The junk mail problem is swearing for the e-mail user who receive over hundreds unwanted emails a day, whether or no longer that junk mail in reality benefits the spammer isn't clearly recognized. Spammer is a person or organization that sends irrelevant or undesirable messages over the net. If user not completely recognize the email that the mail is junk or ham than most of the time is wasted in studying and deleting junk mails. Identification of unsolicited mail is a main trouble at both the commercial and personal end of internet consumer and internet provider.

1.1. Damage Caused by Spam

On the organizational level, spam consequences include: irritation to individual users, less accurate communications, loss of work productivity, misuse of network bandwidth, waste of file server storage space...
and computational power, spread of viruses, worms and Trojan horses, financial losses by phishing, Denial of Service (DoS), directory harvesting attacks [3]. Program which helps the user in labeling the email as spam or ham that an email is worth reading or not, is known as spam filter program. It detects and prevent spam messages to enter in user inbox. On the basis of some criteria spam filter made judgments. Spam may not affect user privacy directly but it ends up occupying large chunk of user inbox capacity.

1.2. Approaches for Email Identification

Classification of emails into categories is a problem. Classification is known as a category into which something is assigned a type. The two approaches used in e-mail identification are knowledge engineering and machine learning. A set of rules must be specified in the knowledge engineering approach, according to which emails are classified as spam or ham [4]. No promising results are shown by implementing this approach because the rules must be constantly updated and preserved, which is a waste of time and is not convenient for users. Machine learning is more efficient than the approach to knowledge engineering. [1]. Instead, a set of training samples used, these samples are a set of labeled pre classified e-mail messages. A specific algorithm is then used to learn the classification rules from these e-mail messages. This phase is known as learning phase.

Machine learning approach has many algorithms used in e-mail filtering. That includes Naïve Bayes, support vector machines, Artificial Neural Networks (ANN), Decision tree J48, K-nearest neighbor, ID3 (Iterative Dichotomiser 3). Results show that Naïve Bayes is the best classifiers against several common classifiers (such as decision tree, neural network, and support vector machines) in term of accuracy and computational efficiency [5]. In this paper Naïve Bayes technique is discussed for classification and pseudo code of classifier for performing text classification of emails. Feature extraction is important task in the model training, text feature extraction techniques are discussed in the paper.

2. Literature Review

Emails are one of the main communication method for personal and professional internet user. In 1999 first anti-unsolicited email filters had been created to protect personal level users and communities from receiving spam mail [5]. Authors’ work is listed here to classify ham and spam documents. Table 1 illustrates the comparative work of authors with the aid of pointing out the classification strategies, year of publication, problem, dataset used, approaches, classifier used and accuracies they achieved.

This Xiu-Li Pang, Yu-Qiang Feng and Wei Jiang [6], [7] have proposed a spam email filtering system that uses four different feature selection method to classify emails. Specialty word extraction algorithm based on information entropy is presented in paper which results in increased accuracy. They considered content-based approaches using English and Chinese email datasets. Naïve bayes and SVM classifiers is used for classification. SL Ting et al [5] presented another machine learning algorithm based on the work on the classification of text data. The implementation of the algorithm includes Naïve Bayes, who proposed a system for classifying documents into different categories. Features are selected by Cfs subset evaluator and rank search. Naïve bayes achieved best accuracy 97% on 4000 document dataset, they proved that naïve Bayes is a good classifier for text classification. Further, WA Awad et al [4] proposed a content based email filtering method. They discuss different classifier method including Naïve Bayes, k-nearest neighbor, ANN, SVM, AIS, Rough set classifier method. Performance of different classifier are compared on 6000 datasets. Naïve Bayes achieved best accuracy of 99%. Authors also survey some popular filtering machine learning algorithm that rely on text classification. Saab et al [8] did comparative study on SVM, LM-SVM, DT and ANN classifiers on spam base dataset of 4597.

Use Another illegal mail recognition method was suggested by Tarjani Vyas et al [9]. The goal is not only to identify mail as spam or ham, but also to minimize the risk of false positive and false negative. They
considered 1020 mails collected from members of their lab. For feature extraction TD-IDF is computed for each message word. Machine learning methods used are Clustering, ID3, J48, NB, SVM and ANN and they achieved 93% accuracy. Authors also work on features selection based on improved mutual learning algorithm. Liang Ting et al [10] proposed improved mutual information method with average frequency of feature selection system. Experiments are conducted based on the English corpus and Chinese corpus CCERT email dataset, the features are extracted through the improved algorithms, and the mails are classified by the Naïve Bayes algorithm. Its select better features and enhance classification results. M.Deepika et al [3] proposed system that check performance of different classifiers with different dataset sizes. Classifiers considered are Bayes Theorem, Naïve Bayes, Rough Set Theory, K-Neighourest Neighbor, Support Vector Machines, Neural networks and Decision Tree. Content based approach is used in their paper. Alurkar Aakash Atul et al [11] proposed program attempts to use machine learning techniques to detect a pattern of repetitive keyword architecture through which a spam filter can be introduced. Parameters such as-To field, from field, Message-ID, Cc / Bcc field, etc. were used from the email header. The paper also takes into account the email body of widely used keywords and punctuations. They used both the content and header based approach.

Recently, comparison between different machine learning classifiers are used for spam classification, is presented by Nasreen M Shajideen [12]. They have discussed the effect of spam and classifiers used for comparison are SVM, NB and J48. Enron1 dataset from Enron spam have been used by them and they achieved 94% accuracy using SVM classifier.

3. Approaches Used for Email Spam Detection

An electronic mail contains of different parts and fields. For classification purpose those parts are considered which are more helpful for correct prediction. For these classification two approaches are defined for spam detection. One is content based approach in which the content of the emails is extracted for the classification purpose, body of the email is main content of the email, actual spamming material lies in body of the email.

The second approach used for email spam detection is header based approach, in which the fields other than body of emails are extracted for classification. The fields which are considered are to (receiver of email), from (sender of email), Cc (carbon copy) / Bcc (Blind carbon copy), Message-ID, return path, IP addresses, subject of the email. Phishing material also exist in the body of the email, from our literature review we come to know that many researcher work on content based approach. So, we also work on Content based approach.

4. Text Feature Selection

When dataset is in text form the feature selection is different from numerical data. If we want to apply machine learning on a text data, then firstly we have to transform the string into numerical feature vector. Because we need numerical feature as input for our algorithm. Word frequency can be chosen as text feature. To make classifier more advance, TF-IDF, N-grams, lemmatizing etc. can be chosen [13].

| Author/Year       | Problem          | Approach                  | Classifier | Accuracy                          | Data Set               |
|-------------------|------------------|---------------------------|------------|-----------------------------------|------------------------|
| Xui-li Pang et al. [7] 2007 | Spam problem     | Content based (subject line + mail body) | Naïve bayes SVM | Specialty words can increase accuracy by 1.8 percent. | 2893 messages 2412 legitimated 481 spam 4000 instances of dataset |
| SL Ting et al. [5] 2011 | Documents Classification | Content based | Naïve bayes | 97% accuracy achieved by Naïve Bayes |
4.1. Bag of Word Representation

It is an easiest way of using bag of word representation. All words must be extracted from all the training dataset. It will build dictionary of the words. Each word is referred to an index which is a integer number, and then count the occurrence of each word in the document, this build a feature vector with word counts. Word frequency is chosen as the text feature.

4.2. Removing Stop Words

The words which did not add meaning to classification process such as and, a, ever and so on. These are stop word; each language has different stop words. These word should be removed from dataset.

4.3. Using N-gram

It doesn’t count single word but instead of it, sequence of word is count. It can capture phrase and expression of text. It’s like a sliding window which move around the word. It has specified length in which continues sequence of character is considered, like “hacked account” and “get money”.

4.4. TF-IDF

TF stands for term frequency. IDF stands for inverse document frequency. Its not only count the word but did something more advance by complimenting frequently occurred words in documents. Its re-weight the count features.

4.5. Lemmatizing Words or Words Stemming

Different form of same words is grouped together. Words are reduced to their stem. So eat, ate, eating and eaten are considered as same.

5. Naïve Bayes

Naïve Bayes is a machine learning classifier, which is probabilistic classifier that is good for text classification. The naive Bayes algorithm is called "naïve" because it assumes that the occurrence of one
feature is independent of the occurrence of other features. This refers to the statistician Thomas Bayes and the theorem named after Bayes’ theorem, which is the base of the naive Bayes algorithm [14]. The Naïve Bayesian classifier is based on Bayes theorem which is the strong independence assumptions between the features. It is based on statistical technique probability for email filtering.

It’s a machine-learning algorithm for supervised learning. It is used mainly for text identification, which requires high-dimensional learning datasets. Topics include spam filtration, sentimental evaluation and the identification of news articles. The Naïve Bayes algorithm is based on conditional probability. The probability that event A will occur, provided that event B has occurred, is called the conditional probability. The conditional probability of A Given B, is denoted by the symbol \( P(A \mid B) \). Formula of conditional probability is by using (1)

\[
P(A \mid B) = \frac{P(A \cap B)}{P(B)} P(A \cap B)
\] (1)

Conditional probability is the probability of event A given that event B has occurred is equal to probability of event A Intersection B over probability of event B as in [15].

5.1. Bayes Theorem

Theorem of Bayes using (2) is defined as the probability of event A given that B is equal to the probability of event B given multiplied by the probability of event A on the probability of event B.

\[
P(A \mid B) = \frac{P(B \mid A)P(A)}{P(B)}
\] (2)

A is called the proposition and \( P(A) \) is called the prior probability of proposition and \( P(B) \) is called the prior probability of evidence. \( P(A \mid B) \) is called the posterior. \( P(B \mid A) \) is the likelihood. Equation (3) is the real form of equation [16].

\[
\text{Posterior} = \frac{\text{likelihood*proposition}}{\text{Evidence}}
\] (3)

5.2. Derivation of Bayes Theorem from Conditional Probability

Equation (4) is the formula for conditional probability by using this we can derive Bayesian theorem

\[
P(A \mid B) = \frac{P(A \cap B)}{P(B)} \Rightarrow P(A \mid B).P(B) = P(A \cap B)
\] (4)

\[
P(B \mid A) = \frac{P(B \cap A)}{P(A)} \Rightarrow P(B \mid A).P(A) = P(B \cap A)
\] (5)

By using (4) and (5), as we know that \( B \cap A \) and \( A \cap B \) are equal so probabilities of these event are also equal. We get equation (6) as follow:

\[
P(B \mid A).P(A) = P(A \mid B).P(B)
\] (6)
The main aim in the Naïve Bayes algorithm is to calculate the conditional probability of an object with a feature vector \( x_1, x_2, \ldots, x_n \) belongs to a particular class \( c_i \) in (7).

\[
P(C_i | x_1, x_2, \ldots, x_n) = \frac{P(x_1, x_2, \ldots, x_n | C_i).P(C_i)}{P(x_1, x_2, \ldots, x_n)} \quad \text{for} \quad 1 \leq i \leq K
\]  

(7)

5.3. **Pseudo Code for Email Classification Using Naïve Bayes**

**Stage 1: Training**
1. Extract email text from dataset
2. Parse each email in tokens
3. The training dataset consisting of documents belongs to different classes, i.e. class \( A \) and class \( B \). Calculate the class’s prior probabilities.
   \[
P(C_i) = \frac{\text{number of objects of class}}{\text{Total number of objects}}.
\]
   \( C_i \) can be \( A \) or \( B \) as \( P(A) \) and \( P(B) \) by same formula.
4. Generate a frequency of for each token \( W \) with respect to classes as \( x_i \).
5. Find ad, all the unique word and count in the whole training dataset document.
6. Count number of words in each class as \( N_c \).

**Stage 2: Filtering**
7. Classify a new document or email \( N \) based on the \( P(C | N) \) i.e. \( C \) is class we have two classes \( A \) and \( B \). \( N \) document contain words of \( N_1, N_2, N_3, \ldots, N_n \).
   a) \( P(A | N) = P(A) * P(N_1 | A) * P(N_2 | A) * \ldots * P(N_n | A) \)
   b) \( P(B | N) = P(B) * P(N_1 | B) * P(N_2 | B) * \ldots * P(N_n | B) \)
8. Use Laplace smoothing to avoid zero frequency problems. In Laplace smoothing method each word frequency is added by 1. If zero frequency occur by adding one zero problem will minimize. The estimator is explained below using (8).
   \[
P(\text{word}N | C) = \frac{\sum_{i=1}^{d} x_i + 1}{N_c + d} (i = 1, \ldots, d)
\]
   (8)

Calculate this for both classes \( A \) and \( B \) as \( NcA, NcB \)
9. Assign a new e-mail label to the higher probability the class has achieved. If \( P(A | N) \geq P(B | N) \) than label class \( A \) else class \( B \).

6. **Experiment**

In today’s electronic world spam classification is major issue. Different spam classification methods are used to solve spam problem. In the mailbox spam and non-spam mails identification is done by using spam detection technique. For spam detection different steps are followed. To implement classification classifier, a model is trained on each training set and performance of model is checked on testing dataset. The model we build based on “Naïve Bayes” classifier is trained in WEKA, (Waikato Environment for Knowledge Analysis) software that was developed at the University of Waikato in New Zealand. WEKA tool supports to a wider range of algorithms & very large data sets [17]. And we also build model on “Naïve Bayes” using web based system built utilizing php-ml library [18].

In addition, the tests were carried out in compliance with the following environmental specifications:
- Processor: Intel Core i7 CPU 860@2.5 GHz 2.93Hz.
• Operating System: Windows 10, 64-bit.
• Memory (RAM): 8.00 GB.

6.1. Installing PHP-ML

After downloading the composer for dependency manager. A file name composer.json was created in the root directory of the project. Using command line composer is installed in the selected project directory, library name and version is mentioned in the file. After successful installation of PHP-ML composer. lock named file is shown in the directory.

6.2. Text Classification

A dataset with 960 emails is used for the modeling training and testing. Dataset is taken from Git hub [19]. The selected data set contain two categories of document: ham and spam. All the two categories are easily differentiated. 27% data (i.e. 260 emails) are extracted randomly to build the testing dataset for the classifier. The other 700 emails are used as the training dataset to train the classifier. 1557 attributes are selected.

Email dataset is preprocessed and it's done for precise and good features acquisition. Email data is extracted and it text is converted into numerical form by tokenizing the text and numeric representation. It is necessary for model training to convert text into numeric values, stop words are remove and tf-idf of document is calculated. For refining the dataset, to achieve good results. Bag of word representation of text, N-gram technique which is work as sliding window for semantic accuracy it has variable length. In weka we use experimental explorer for model training. We select supervised learning, string to word vector function on dataset. Then stop word removal, Tf-idf count and set attribute as numeric class. After this preprocessing, classifier is chosen we choose naïve Bayes and perform splitting of data for training and testing.

The training is based on a Naïve Bayes model on each of the training set. After training the performance of model is tested on testing dataset. The resultant accuracy of the model is given first one is the accuracy achieved by the web based email management system result which detect spam emails in real life application. And the second results are of model built on Weka. We also calculated model accuracy by using SVM as classifier on same dataset by looking upon the experimental result. Because in many research paper SVM also gives good results.

6.3. Performance Matrices

We use following standard matrices to evaluate the proposed system: Fmeasure, accuracy, recall and precision using publicly available dataset. The correctness of a classification can be evaluated by computing the number of correctly recognized class examples (true positives), the number of correctly recognized examples that do not belong to the class (true negatives), and examples that either were incorrectly assigned to the class (false positives) or that were not recognized as class examples (false negatives) [20].

6.3.1. Accuracy

Accuracy is characterized by number of right predication partitioned by complete number of predications

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}
\]  

as the equation written in (9) it's characterize by and large viability of model.

6.3.2. Recall (Sensitivity)

Recall can be characterized as the proportion of the all-out number of effectively grouped positive models gap to the all-out number of positive models. High Recall shows the class is effectively perceived (modest number of FN). Recipe of review in (10) is characterized.
Recall = \frac{TP}{TP + FN} \quad (10)

Effectiveness of classifier to recognize positive labels. on the off chance that we have a 100% sensitivity model, that implies it didn't miss any Evident Positive, as it were, there were no false Negatives (for example a positive outcome that is named as negative). However, there is a danger of having a great deal of false Positives.

6.3.3. Precision

To get the estimation of precision we separate the all-out number of effectively arranged positive examples by the absolute number of anticipated positive examples. High precision shows a model marked as positive is to be sure positive (modest number of FP). Class understanding of the information names with the positive names given by the classifier. In (11) formula of Accuracy is given by the connection.

\text{Precision} = \frac{TP}{TP + FP} \quad (11)

6.3.4. F-measure

Since we have two measures (Precision and Recall) it has an estimation that speaks to them two. We ascertain a F-measure, which utilizes Harmonic Mean instead of Mean as it rebuffs the extraordinary qualities more. formula for F-measure is shown in (12). The F-Measure will consistently be closer to the littler estimation of Precision or Recall.

\text{F - measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (12)

6.4. Naïve Bayes and SVM Evaluation Measure

Table 2 shows the al evaluation measures of our web based system in which accuracy, precision, recall and f-measure is mentioned which is achieved by our system. The chart for its comparison in Fig. 5 has been displayed. Results shows naïve Bayes performed better than SVM.

| Table 2. Evaluation Measures |
|------------------------------|
| Metrics | Naïve Bayes | SVM |
| Accuracy | 0.97 | 0.9 |
| Precision | 0.97 | 0.91 |
| Recall | 0.95 | 0.9 |
| F-measure | 0.97 | 0.85 |

Fig. 1. Evaluation measure.
6.5. Comparison of our Web-Based System Using Naïve Bayes with Previous Systems

The model we built using naïve Bayes algorithm utilizing php-ml library. Our proposed system has achieved 97% accuracy. We have compared our built model results with the previous systems from which our system performed better in classification of document, table II presents summary of results. And analysis of result is mentioned in Fig. 1 in the form of comparison chart. In 2011 SL ting et al. achieved 96% accuracy by their system. So we take their Naïve Bayes accuracy results to compare with our results. W.A et al. used 6000 datasets and they built model on many classifier and achieved 96% accuracy. So we also take their naïve Bayes result to compare with our results. In 2015 Tarjani Vyas et al. used 1020 mails from lab member and built a model on many classifiers achieved 91% accuracy using naïve Bayes by their system. In 2018 Nasreen M Shajideen. et al. used 3762 spam, 3172 ham emails and achieved 92% accuracy.

Table 3. Comparison with Previous System

| Metrics    | Proposed system | SL[4] 2011 | W.A[3] 2011 | Vyas[8] 2015 | N.M[11] 2018 |
|------------|-----------------|------------|-------------|--------------|--------------|
| Accuracy   | 0.97            | 0.96       | 0.96        | 0.91         | 0.92         |

Fig. 2. Comparison with previous systems utilizing Naive Bayes.

6.6. Comparison of Results Generated by Proposed System Using Naïve Bayes on Weka, with Previous System

We built model using naïve Bayes algorithm utilizing Weka and achieved 95% accuracy. We compare our Weka build model result in Table 3 with the previous systems 2015 Tarjani et al. and 2018 Nasreen et al. and in chart Fig. 2 analysis of these comparison result is presented for visual representation. We compare our results with two systems from which our system performs best.

Table 4. Comparison with Previous System

| Metrics    | Proposed system | Vyas[8] 2015 | N.M[11] 2018 |
|------------|-----------------|--------------|--------------|
| Accuracy   | 0.97            | 0.91         | 0.92         |

Fig. 3. Comparison of Naive Bayes using Weka.
6.7. **Comparison of Results Generated by Proposed System Using Naïve Bayes and SVM Classifier**

We compare our Naïve Bayes model build results with SVM (Support Vector Machine). SVM is also a good classifier but for text classification Naïve Bayes classifier works more precisely and it is proved from the results presented in Table IV in which Naïve Bayes performed better than SVM. And Fig. 3 their comparison chart is displayed for better understanding.

| Metrics | Naïve Bayes | SVM |
|---------|-------------|-----|
| Accuracy | 0.97 | 0.90 |

![Comparison Chart](chart.png)

Fig. 4. Naïve Bayes and SVM comparison.

7. **Conclusion and Future Work**

This paper was presented for an in-depth knowledge of the naïve Bayes algorithm, which is the best text classification algorithm. Its formula and pseudo code explained, and for more advanced results the selection of different features is discussed. We create our own web-based system, and we also perform WEKA calculation. Naïve Bayes works best with a high dimension. Naïve Bayes provides quick and effective performance. We observed that naive Bayes model training takes 9.2s and SVM takes 28.7s.

Both parts of the email can be considered to create a more reliable spam mail scanning system. Header-based approach will be viewed with a content-based approach to classification. The efficiency of the algorithm is discussed in the spam email classification field. In model training diverse and broad datasets will be used.

**Conflict of Interest**

Authors declared that there is no conflict of interest.

**Author Contributions**

The basic idea was proposed by Mr. Mohsen Ansari and guided throughout the implementation of this web-based system. This web-based spam email filtering system have been developed by Ms. Asma Bibi and Rasia Latif with the help of Mr. Ehtsham Shabir and Mr. Waqas Ahmed under the supervision of Dr. Samina Khalid and Dr. Tehmina Shehrya.

**Acknowledgment**

We would like to thank all those who helped us in completion of this research work.
References

[1] Christina, V., Karpagavalli, S., & Suganya, G. (2010). Email spam filtering using supervised machine learning techniques. *International Journal on Computer Science and Engineering (IJCSE)*, 2, 3126-3129.

[2] Cyber Attack Works. Retrieved from https://www.csoonline.com/article/2117843/what-is-phishing-how-this-cyber-attack-works-and-how-to-prevent-it.html

[3] Deepika, M., & Shilpa, R. *Performance of Machine Learning Techniques for Email Spam Filtering.*

[4] Awad, W. A., & ELseuofi, S. M. (2011). Machine learning methods for spam e-mail classification. *International Journal of Computer Science & Information Technology (IJCSIT)*, 3(1), 173-184.

[5] Ting, S. L., Ip, W. H., & Tsang, A. H. (2011). Is Naive Bayes a good classifier for document classification. *International Journal of Software Engineering and Its Applications*, 5(3), 37-46.

[6] Background of Spam. Retrieved from http://www.articles-about-spam.com/background-of-spam.shtml

[7] Pang, X. L., Feng, Y. Q., & Jiang, W. (2007). A spam filter approach with the improved machine learning technology. *Proceedings of Third International Conference on Natural Computation (ICNC 2007): Vol. 2* (pp. 484-488).

[8] Saab, S. A., Mitri, N., & Awad, M. (2014). Ham or Spam? A comparative study for some content-based classification algorithms for email filtering. *Proceedings of MELECON 2014-2014 17th IEEE Mediterranean Electrotechnical Conference* (pp. 339-343).

[9] Vyasa, T., Prajapat, P., & Gadhwal, S. (2015). A survey and evaluation of supervised machine learning techniques for spam e-mail filtering. *Proceedings of 2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (pp. 1-7).

[10] Ting, L., & Qingsong, Y. (2012). Spam feature selection based on the improved mutual information algorithm. *Proceedings of 2012 Fourth International Conference on Multimedia Information Networking and Security* (pp. 67-70).

[11] Alurkar, A. A., Ranade, S. B., Joshi, S. V., Ranade, S. S., Sonewar, P. A., Mahalle, P. N., & Deshpande, A. V. (2017). A proposed data science approach for email spam classification using machine learning techniques. *Proceedings of 2017 Internet of Things Business Models, Users, and Networks* (pp. 1-5).

[12] Shajideen, N. M., & Bindu, V. (2018). Spam filtering: A comparison between different machine learning classifiers. *Proceedings of 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 1919-1922).

[13] Text data classification with BBC news article dataset. Retrieved from https://arkadiuszkondas.com/text-data-classification-with-bbc-news-article-dataset/

[14] Make AI real: Operationalize data science. Retrieved from http://www.statsoft.com/textbook/naive-bayes-classifier

[15] Berrar, Daniel. "Bayes' theorem and naive Bayes classifier." Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics (2018): 403.

[16] Naive Bayesian. Retrieved from https://www.saedsayad.com/naive_bayesian.htm

[17] The workbench for machine learning. Retrieved from https://www.cs.waikato.ac.nz/ml/weka/

[18] Spam Filter Machine Learning. Retrieved from https://github.com/Gago993/SpamFilterMachineLearning

[19] The breadth of machine learning. Retrieved from https://bruceyanghy.github.io/posts/machine_learning_breadth/index_breadth.html

[20] Santra, A. K., and C. Josephine Christy. "Genetic algorithm and confusion matrix for document clustering." *International Journal of Computer Science Issues (IJCSI)* 9, no. 1 (2012): 322.
Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).

Asma Bibi just have completed her MCS degree from the Department of Computer Science and Information Technology, MUST, Mirpur AJK, Pakistan. She is looking forward to continue her graduate studies. Her field of interest is web development, and machine learning.

Rasia Latif just have completed her MCS degree from the Department of Computer Science and Information Technology, MUST, Mirpur AJK, Pakistan. She is looking forward to continue her graduate studies. Her field of interest is databases, web development, and machine learning.

Samina Khalid is serving as a lecturer in the Department of Computer Science and Information Technology, MUST, Mirpur AJK, Pakistan. She has honors degree in computer sciences and master's degree in software engineering. She has recently completed her Ph.D in software engineering. Her interest areas are biomedical image processing, machine learning, and pattern recognition.

Waqas Ahmed just have completed her MSCS degree from the Department of Computer Science and Information Technology, MUST, Mirpur AJK, Pakistan. He is serving as a lecturer in the Department of Computer Science and Information Technology, University of Kotli, Kotli AJK, Pakistan. His field of interest is neural networks, digital image processing, and machine learning.

Raja Ahtsham Shabir is pursuing his bachelor’s degree in information technology from the Department of Computer Science and Information Technology, MUST, Mirpur AJK, Pakistan. His field of interest is web development, clustering, and machine learning.

Mohsin Ansari is serving as a lecturer in the Department of computer science and information technology, MUST, Mirpur AJK, Pakistan. He has completed his master’s degree in computer science from Qaid-e-Azam University, Islamabad, Pakistan. He has knee interest in web technologies, programming, and machine learning.

Tehmina Shehryar is serving as a lecturer in the Department of Software Engineering, MUST, Mirpur AJK, Pakistan. She has honors degree in computer sciences and master's degree in software engineering. She has recently completed her PhD in software engineering. Her interest areas are biomedical image processing, machine learning, and pattern recognition.