Multimedia Tools and Applications (2023) 82:29227–29254
https://doi.org/10.1007/s11042-023-14689-3

Comparison of machine learning techniques for spam detection

Argha Ghosh [1] · A. Senthilrajan [1]

Received: 14 December 2021 / Revised: 7 April 2022 / Accepted: 4 February 2023 /
Published online: 20 February 2023

Abstract
Email is a useful communication medium for better reach. There are two types of emails, those are ham or legitimate email and spam email. Spam is a kind of bulk or unsolicited email that contains an advertisement, phishing website link, malware, Trojan, etc. This research aims to classify spam emails using machine learning classifiers and evaluate the performance of classifiers. In the pre-processing step, the dataset has been analyzed in terms of attributes and instances. In the next step, thirteen machine learning classifiers are implemented for performing classification. Those classifiers are Adaptive Booster, Artificial Neural Network, Bootstrap Aggregating, Decision Table, Decision Tree, J48, K-Nearest Neighbor, Linear Regression, Logistic Regression, Naïve Bayes, Random Forest, Sequential Minimal Optimization and, Support Vector Machine. In terms of accuracy, the Random Forest classifier performs best and the performance of the Naïve Bayes classifier is substandard compared to the rest of the classifiers. Random Forest classifier had the accuracy of 99.91% and 99.93% for the Spam Corpus and Spambase datasets respectively. The naïve Bayes classifier had the accuracy of 87.63% and 79.53% for the Spam Corpus and Spambase datasets respectively.

Keywords Machine learning algorithms · Classification · Spam email detection · Machine learning · Artificial intelligence

1 Introduction

In modern life, Email is the best medium for formal communication. Moreover, Email is the easiest way to communicate. Generally, there are two types of emails, those are ham or

✉ Argha Ghosh
argha.ghosh16@gmail.com

A. Senthilrajan
agni_senthil@yahoo.com

[1] Department of Computational Logistics, Alagappa University, Karaikudi, India

legitimate email and spam email. An Email contains two parts, those are email body and email header. But, at present time Email has been misused in the name of "Spam". Spam is also a kind of bulk or unsolicited email that contains an advertisement, phishing website link, malware, Trojan, etc. we all used to receive a lot of emails in a day; out of which, 70%–80% of emails are spam. Spam emails are used to send by spammers with many intentions like hacking, phishing, banking fraud, etc. Social media is the best medium for Spammers for getting the personal data of the user by sending spam emails. Spam is also used to know as "junk emails". Spam emails are used for content advertisement, offers, phishing website links, anonymous virus-like malware, trojan, etc. "SPAM" is derived as Self Propelled Advertising Material [97]. In 2019, worldwide more than 280 billion spam emails are been sent and received. According to Google, 64% of emails sent and received in 2019 are spam emails and, this rate used to increase every year by 2%–3%. There are two types of spam detection techniques are there. These are sender based spam detection and content-based spam detection [33]. Sender based spam detection mainly happened based on features like Content-Type, Message-ID, MIME-Version, Authentication-Results, and Return-Path [12]. In content-based spam filtering, it checks the text of an email's message as well as checks the URL of the email with the subject of the email for text classification [86]. In this research work, content-based spam detection has been done. There are three types of spam filters are there, for filtering spam emails, those are Blacklist Filter, Whitelist Filter, and Content-based Filter [6].

The term Machine Learning (ML) defines that the machine learns the characteristics cum behaviour from experience; it's an application of Artificial Intelligence (AI). Machine Learning is generally classified into three types; those are Supervised Learning, Unsupervised learning and, Reinforcement Learning. There are various machine learning classifiers are there based on a particular algorithm. In this research work, thirteen machine learning classifiers have been implemented. These are Adaptive Boosting, Artificial Neural Network, Bootstrap Aggregating, Decision Table, Decision Tree, J48, K-Nearest Neighbor, Linear Regression, Logistic Regression, Naïve Bayes, Random Forest, Sequential Minimal Optimization and, Support Vector Machine for detecting spam emails from two datasets. Those two data sets are Spam Corpus (http://lpis.csd.auth.gr/mlkd/spam_corpus2.rar), and Spambase (https://archive.ics.uci.edu/ml/datasets/Spambase). Waikato Environment for Knowledge Analysis [30] is open-source software for performing the task of data mining operations like Pre-processing, Classification, Clustering, etc. It was invented by The University of Waikato, Hamilton, New Zealand in 1999. In this research work, WEKA 3.9.4 version has been used.

This research paper is organized as follows. The second section will contain the related work or existing research work on spam email detection using machine learning algorithms. The third section will present all the implemented machine learning classifiers. The fourth section will present the datasets. The fifth section will be delivered the spam detection approach. The sixth section will derive experimental analysis. The seventh section will conclude this research work with a future aspect.

2 Related work

This section will derive the existing research on spam email detection using machine learning classifiers. This survey focused on all those classifiers that were used in past for spam detection. This survey focused on classifiers that are used and which one has the best accuracy. In Table 1 details of the survey have been given below.

Table 1 Literature review of spam detection using machine learning classifiers

Authors	Classifier Used	Best Classifier (Accuracy)
Konstantin Tretyakov [105]	Naïve Bayes, k Nearest Neighbors, Artificial Neural Network, Support Vector Machine	Artificial Neural Network (98.5%)
Ali Shafigh Aski et al. [91]	Naïve Bayes, J48, Multi-Layer Perceptron	Multi-Layer Perceptron (99.3%)
Jose R. Mendez et al. [62]	Naïve Bayes, SVM, C4.5, Adaboost C4.5, Bagging C4.5, Random Forests, Logistic Regression, Rough Sets	Rough Sets (99.4%)
Abdulhamit Subasi et al. [98]	C4.5, CART, REP Tree, LAD Tree, NB Tree, Random Forest, Rotation Forest	Random Forest (95.80%)
Prachi Gupta et al. [36]	Naïve Bayes, Support Vector Machine	Naïve Bayes (99.49%)
Muhammad Ali Hassan et al. [37]	Naïve Bayes, Support Vector Machine	Support Vector Machine (99%)
Frank Vanhoenshoven et al. [109]	Decision Trees, k-Nearest Neighbor, Bayesian Networks, Random Forest, Support Vector Machine, Multi-Layer Perceptron	Random Forest (97.69%)
Shafi'i Muhammad Abdulhamid et al. [1]	Bayesian Logistic Regression, Hidden Naïve Bayes, RBF Network, Voted Perceptron, Lazy Bayesian Rule, Logit Boost, Rotation Forest, NNge, Logistic Model Tree, REP Tree, Naïve Bayes, J48, Multilayer Perceptron, Random Tree	Rotation Forest (94.2%)
Prabin Kumar Panigrahi [71]	Artificial Neural Network, Support Vector Machine, Random Forest	Random Forest (99.93%)
Amani Alzahrani et al. [4]	Logistic Regression, Naïve Bayes, Support Vector Machine, Neural Network	Neural Network (97.67%)
Yuliya Kontsewaya et al. [51]	k-Nearest Neighbor, Naïve Bayes, Decision Tree, Support Vector Machine, Random Forest, Logistic Regression	Naïve Bayes, Logistic Regression (99%)
Mahmoud Bassiouni et al. [7]	Artificial Neural Network, Bayes classifier, Decision Table, K-Nearest Neighbor, Logistic Regression, Naïve Bayes, Radial Basis Function, Random Tree, Random Forest, Support Vector Machine	Random Forest (95.46%)

3 Classifiers

3.1 Adaptive booster

Adaptive Booster (AdaBoost) is a boosting algorithm and a well-known ensemble method in Machine Learning. The term "ensemble" method defines that the classifier takes the results of multiple classifiers together and mixes them for better accurate results. AdaBoost was invented by Yoav Freund and Robert Schapire. AdaBoost is a machine learning classifier used for finding targets by training different classifiers for the same training set to make a powerful classifier [87]. In other words, AdaBoost merges a branch of weak learners to make a single learner that is stronger than a particular learner [32]. AdaBoost is used for object detection in classification. AdaBoost is the first boosting algorithm and also a meta-algorithm from the perspective of machine learning. Generally, AdaBoost can be implemented in three ways; those are using Haar-like features, introducing Local Binary Pattern (LBP), and implementing Field Programmable Gate Arrays (FPGA) [44]. AdaBoost has two special features in terms of classification; those are interpolation and generalization of errors respectively in the process of classification. The AdaBoost classifier is briefly explained in [90].

3.2 Artificial neural network

An Artificial Neural Network is known for information processing which has been the collection of interconnected neurons and is inspired by the human nervous system [35, 92]. The structure of an ANN consists of one input, one output and one or more hidden layers in between [114]. It's difficult to predict how many numbers of hidden layers and hidden neurons are present in between the input and output layers of an ANN. In terms of a large number of classification, clustering, regression, pattern recognition and prediction, Artificial Neural Network is very successful in many disciplines [2, 68]. In an ANN, given input is used to go next layer and every layer has connections. And, each connection has associated weight. Input values are used to get multiplied with associated weights of connection and summarised to form a new input for the neuron. And, the new input achieves through an activation function. Each neuron of a network used to have a nonlinear activation function [124]. There are seven types of ANN; those are Modular neural network, Recurrent neural network, Generative neural network, Deep neural network, Spiking neural network, Feed-forwarded neural network and Physical neural network [18]. Perceptron is the most known architecture of neural networks [107].n

$$y(x) = g\left(\sum_{i=0}^{n}(w_i x_i)\right) \tag{1}$$

In Eq. 1, x_i is the i^{th} neuron in the previous layer; w_i is the weight deciding parameter that would decide the weight of the neuron; g is the activation function.

3.3 Bootstrap aggregating

Bagging is a powerful widely used ensemble classifier in Machine Learning (ML) and is known as a meta-estimator in terms of classifying datasets. Bagging is derived as "Bootstrap Aggregation". Bagging makes bootstrap data sets for replacing the actual dataset [39]. Bagging is used to reduce the variance where the dataset contains high variance. In the case of spam email classification, bagging is used for counting the spam functional words in terms of occurrences for training the dataset [19]. Bagging performs n number of classifications based on bootstrap sampling data of training dataset. Lastly, conquers all the results in one as a final prediction. The bagging classifier is fully explained as well as elaborated in [15].

3.4 Decision table

In our research work, Decision Table is the fourth classifier used for detecting spam emails. In terms of numerical prediction, Decision Table is the best accurate classifier for forming decision trees [46]. The decision Table represents the visual model of the classification process in terms of tables with attributes of actual data. Decision Tables can take different decisions or actions based on a set of conditions [119]. Decision Table can easily maintain the data with different versions cum order in terms of classification. The structure of the Decision Table looks like a relational table, in that table, each row contains aggregate, combinations of values, attributes, etc. [8]. Decision Table is popular in terms of classifiers because it's easy to understand from an overview itself. The decision Table contains two sub-classifier and those are DTMaj (Decision Table Majority) and DTLoc (Decision Table Local) [50]. Decision Table Majority has two components; those are schema and body [49]. Decision Table used to

hold more data than the top, mainly following the tree kind of structure. Data of the dataset has been divided and constructed decision table by the decision table classifier [48]. From those various decision tables based on attributes, a decision tree classifier is used to make decisions.

3.5 Decision tree

A Decision Tree is a supervised machine learning technique used for classification and regression. A Decision Tree can be formed using a set of instances through the divide-and-conquer paradigm [84]. A Decision Tree is a supervised tree where internal nodes are testing nodes and leaf nodes are decision nodes [31, 70, 75]. Algorithms like Classification and Regression Tree (CART), Iterative Dichotomiser (ID3), and Chi-Squared Automatic Interaction Detector (CHAID) are useful for creating decision trees. The Decision Tree is useful for clearness and understandability [121]. The decision tree performs the task of finding which attributes will select from each level. Without changing in core logic, a decision tree can scale easily from linear data to non-linear data [43]. A Decision Tree is a graphical representation of all possible solutions to a problem based on given conditions. A Decision Tree is also specified as a hierarchical classifier because it wants multi-level prejudice to decide which class a specific pattern belongs to [117].

3.6 J48

J48 algorithm was invented and developed by Ross Quinlan, and it's also known as the C4.5 algorithm. The C4.5 algorithm was earlier known as the ID3 algorithm. Moreover, the C4.5 algorithm is an extension of the ID3 algorithm. The ID3 algorithm was also invented by Ross Quinlan. J48 usually form the decision tree by the attributes of the training set [72]. J48 algorithm is used to construct a decision tree for classification. The decision tree of J48 looks like a graph that contains a branching method to show every possible outcome of the decision [65]. By seeing the decision tree of J48, we can predict the approximate outcome of classification as well as it helped to understand the classification. J48 have an advantage like finding missing values, pruning of decision tree, ranges of a continuous attribute value, and rules of derivation [85]. The outcome of the J48 classifier is the combination of multiple decision trees; J48 produces the result by conquering the results of those decision trees. The output of the classification for J48 is always used to present as a binary tree [95]. At the time of classification, the J48 classifier is used to generate a decision tree based on training data, and that's the best part of the J48 classifier for understanding the classifier for anyone. J48 is the better algorithm compared to several other algorithms for classifying spam emails [3]. J48 algorithm can be implemented on devices for classification as well as useful for detecting diseases also. J48 is the new version of the C4.5 algorithm [122]. Compare to other popular machine learning algorithms like Naïve Bayes (NB), and Support Vector Machine (SVM), J48 always performs better in terms of performance measurement parameters in the context of classification. J48 is the binary tree for classifying [103].

3.7 K-nearest neighbor

K-Nearest Neighbor is a supervised machine learning algorithm, basically used for resolving classification problems. KNN is a k-related algorithm because its classification accuracy depends on the value of k [42]. KNN is used to calculate the distance between the

classification point and sample data, then sorted closest k points and, lastly allocates the largest k points as points to be classified. Calculating the distance between training and testing sets using Euclidean distance and Mahalanobis distance method is a general task of a KNN [25]. KNN algorithm is effective in pattern recognition. KNN allocates a test sample to the class which has been voted by k-nearest neighbors in training data [26, 34, 52]. KNN algorithm can't predict fundamental data and due to that reason, it is called a non-parametric algorithm. The conventional k-nearest neighbor algorithm presumes that the training samples are steadily assigned among various classes [99]. KNN can't learn from the training set straight away, but it keeps the dataset and executes the dataset at the time of classification. For that reason, the k-nearest neighbor is known as one of the lazy learning techniques [67]. The advantages of the k-nearest neighbor algorithm are simple, easy to implement, and low error rate [60].

3.8 Linear regression

The term linear regression defines the statistical model that shows the relation between a dependent and an independent variable represented in the form of a line equation. Linear regression is a supervised learning algorithm that predicts a certain sample is under the slope or outside of the slope by drawing a lined margin between the samples. It is used to draw the line based on the value of independent and dependent variables. And, by that slope or line, it's used to classify the samples in terms of their values. The equation of linear regression is easy to understand and, it is used for compromising the capacity between volumetric VAT and anthropometric parameters [57]. There are two types of linear regression, those are simple regression and multiple regression. Multiple regression is a complex kind of linear equation whereas, simple regression is the simpler equation to understand. In the context of numerical prediction, multiple linear regression is easy to implement as well as used in statistical applications [73]. Linear regression is used to deal with complex problems compare to other machine learning algorithms. In linear regression, variable significance is the important element [10]. In simple regression, independent and dependent variables always tried to create something like a correlation but, an exact correlation between those variables was never possible. In simple regression, the linear regression model is trained by all available training data [104].

$$Y = a + bX \tag{2}$$

In Eq. 2, X is a dependent variable, Y is an independent variable, b is the slope of the line and a is the intercept.

3.9 Logistic regression

Logistic regression is a statistical model for prediction and a similar kind of classification technique to linear regression. Logistic regression can be used to model the probability of the sample as true/false. Except that true/false, based on the event or class is used to change like pass/fail, slim/healthy, win/lose etc. Logistic regression is never used to calculate the exact value of the sample; it can only predict whether the sample value is true or false. Moreover, logistic regression and linear regression are similar except for the process of classifying the samples. In logistic regression, data points are used to arrange according to the sigmoid function. Logistic regression is a technique that is used for building a model by using multiple

meteorological variables to predict whether precipitation will occur [63]. Generally, logistic regression is used to implement such a scenario where output used to come in binary (0 or 1). Logistic regression is an important model to perform the prediction in a large dataset where important features are selected based on the properties of attributes [118]. Logistic regression comes under the type of regression analysis and is used in a larger dataset where only two types of samples are there like spam filtering (ham or spam). Logistic regression is used worldwide as a classification algorithm [47]. In general, there are three types of logistic regressions; those are binary logistic regression, multinomial logistic regression and, ordinal logistic regression. Logistic regression is popular in statistical learning and machine learning for classifying datasets cum data [59]. Previously, logistic regression was mainly used to solve the binary classification problem. The output of logistic regression is a text segment that is offensive or non-offensive [77]. Logistic regression also comes under the supervised machine learning algorithm category; it is used in regression, multi-classification and, binary classification.

$$\frac{1}{(1+e^{\text{value}})} \tag{3}$$

This is known as the sigmoid function, and it has been developed by the statistician for the properties of the event or class. In Eq. 3, e is the base of the natural logarithm, and value is the numerical value.

3.10 Naïve Bayes

In the eighteenth century, English mathematician Thomas Bayes discover the 'Bayes' theorem. Based on Bayes Theorem, the Naive Bayes classifier was built that is used for computing the unknown classes [45]. Bayes theorem focused on the probability of two events and their conditional probability. The Naive Bayes classifier's assumption is based on class conditional independence [74]. Naive Bayes is a probabilistic supervised machine learning algorithm that calculates a set of probability on given data set based on counting the frequency and the combination of values. A Naive Bayes classifier is used to utilize the word counts in the Bag of Words (BoW) feature extraction for text classification as well as for having the advantage over classification accuracy [33, 83]. Naive Bayes is a simple and easy algorithm to implement compared to other machine learning algorithms. Except for Support Vector Machine and ID3, the Naive Bayes classifier provides faster results and better accuracy [94]. Except for spam classification, Naive Bayes can be used in sentiment analysis, text classification, cyber-attack detection, real-time prediction, multi-class classification, document classification, natural language processing, etc. Naïve Bayes is an easy model to build and, it's useful for working with large data sets [61]. In general, there are three types of Naive Bayes models; those are Gaussian, Multinomial, and Bernoulli. In terms of classification, the Bayesian classifier has a similar kind of ability to a decision tree and neural network for classifying spam emails [28]. Naive Bayes is the best classifier for classifying text; moreover, text classification has the best accuracy using a Naive Bayes classifier. Naive Bayes is used to training a probability model, and it will give each word a probability of being a suspicious spam keyword for classifying email [106]. The Naive Bayes classifier assumes that each feature has an independent and equal contribution to the outcome. So, the Naive Bayes classifier can't learn the relation between features and it's a disadvantage of the Naive Bayes classifier. In spam filtering, Naive

Bayes, Decision Trees, and Support Vector Machine use the vector space method for classical text categorizing [89].

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{4}$$

Bayes' Theorem states that the variable y and dependent feature vector $x_1,\ldots,x_n$ through that,

$$P(y|x_1, \ldots, x_n) = \frac{P(y)P(x_1, \ldots, x_n|y)}{P(x_1, \ldots, x_n)} \tag{5}$$

3.11 Random Forest

Random Forest is the most powerful supervised ensemble tree-based machine learning algorithm that has been used for classification and regression. As the name suggests, the random forest has been consists of many decision trees [41], which are responsible for information retrieval. Each decision tree in a random forest has a result; those results are used to get votes from the decision trees, and the decision tree with the most votes is used to be the outcome of the random forest. Having many decision trees in the random forest makes the random forest algorithm high robustness and for that random forest has high accuracy [20] compare to other machine learning algorithms. Breiman has been proposed the Random Forest algorithm [16] for improving the Bagging algorithm. The random forest algorithm is good in classification, but regression can't meet the expectation, so random forest is not good for regression tasks. Features of random forest are commendable for the reason it is used worldwide [29]. Random forest algorithm used in prediction, banking sector, stock market, medical science, pattern recognition, etc. Artificial intelligence algorithms are mainly used to solve the problem of classification and regression [11]. The random subspace method and bagging algorithm are combined to create the random forest algorithm. Random features are used to select from the input set by the tree classifier [88]. Random forest solves the problem of over-fitting and it has the scalability and parallelism that's help to classify large datasets with higher dimensions. Decision tree and random forest work in the same way but, there is a difference between these two is random forest uses an ensemble learning approach [27]. The random forest comes under the category of Classification And Regression Tree (CART). It uses the tree voting method for bootstrapped data and preparing instructional data [64]. The random forest has high robustness and due to its robustness, the random forest can classify or be suitable to perform classification in high dimensional large data sets. The Random Forest algorithm combines multiple decision trees for upgrading the performance [69]. In the random forest algorithm, all the decision trees are used to train with the bagging algorithm. Random Forest has been used so much because it's easy to implement [76] and for its diversity. Random forest uses the random feature selection method as a dimensionality reduction technique for feature selection. Neural networks and random forests had some similar characteristics [116]. Random forest was introduced in data mining by Ho in 1995 in the name of the random subspace method. The random forest can classify high-quality results without any hyper-parameter tuning [123]. Compare to other machine learning algorithms, random forest selects the features easily and is used to make a good model for predicting by dimensionality reduction technique, which is the reason behind random forest having good accuracy in classification. A random forest is the combination of multiple decision trees, but none of the

decision trees is related to each other [56]. In random forest for solving regression problems, it is used to calculate Mean Square Error for organizing data as a node.

$$\mathrm{MSE} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} (\mathrm{f_i} - \mathrm{y_i})^2 \tag{6}$$

In Eq. 6, N is the number of data points, f_i is the model returned value, and y_i is the value of the data point.

3.12 Sequential minimal optimization

In 1998, John Platt [81] developed the Sequential Minimal Optimization algorithm. For training Support Vector Machine (SVM), an algorithm is needed which can solve the QP programming of SVM. And, from that aspect, only Sequential Minimal Optimization came into the scenario. Sequential Minimal Optimization (SMO) is a fast method to train SVM as well as state-of-the-art solutions for SVM training [55, 66]. The sequential Minimal Optimization algorithm can be divided into two parts; those are the analytical method and the heuristic method. Analytical methods are used to solve the QP problem. The heuristic method is mainly used to recognize the violating pair. Without QP optimization, SMO can solve the SVM QP problem [9, 80, 82]. Sequential Minimal Optimization algorithm is used to solve the problem rise from Support Vector Machine.SMO is used to decrease the calculation period and acquire refined scaling distinctive compared to the SVM training process [24]. SMO solve the QP problem by dividing it into sub-part due to its complexity and then used to solve sub-parts of QP optimization. SMO is a decay method and utilizes the smallest possible working set which can be upgraded successfully [40].SMO is mainly used so much due to its optimization technique. SMO subsequently collect a pair of training samples for join improvisation, which reduces the usage of memory [53].SMO uses an analytical method for avoiding complex iteration processes. SMO can be used as a decay method for training large data sets [93].SMO algorithm depended on heuristics for choosing the variables for optimizing an objective function. This concept helps the algorithm for performing on a large data set. Sequential Minimal Optimization and Support Vector Machine combine perspectives to reduce memory storage, easy to execute with high accuracy [108]. SMO can handle a large training set because SMO consumes the memory for the training set is linear. SMO needs only linear memory for the training set because it didn't compute a large matrix. SMO has mainly been used to reform only two variables for every co-set [58]. SMO can solve the SVM's QP optimizing problem without any extra matrix storage. SMO derive the whole QP problem into the QP sub-problem and is then used to solve those small QP sub-problems. Lastly, it uses to combine the result of each QP sub-problems into one like the divide and conquer technique. At every stage of QP optimisation, SMO is used to solve the smallest possible optimization problem.

3.13 Support vector machines

In 1992, Boser, Guyon and Vapnik [13] develop Support Vector Machines based on statistical learning as a supervised machine learning algorithm for performing classification and regression tasks [101]. SVM performs the job of separating two classes based on a hyper-plane. First time SVM has been implemented by Vapnik for solving a quadratic optimization problem [112]. Quadratic programming is used to solve mathematical optimization problems

presuming quadratic functions. SVM can be used as developing quadratic problems for training the data set [17]. SVM is used to create a decision boundary for putting new data points in the correct category. In SVM, the best decision boundary is used to call hyper-plane. Generally, SVM processed a set of input data (xi) and predicts (yi) and builds a hyper-plane (H) for separating those classes using a hypothesis space for linear function in high dimensional feature space [5, 21, 23, 54]. In SVM, extreme points or vectors are used to create the hyper-plane. And, those extreme points or vectors are used to call support vectors. SVM is always used to maximize the margin between two classes with the help of support vectors [100, 113]. In SVM, the distance between hyper-plane and vectors is known as margin. And, which hyper-plane has maximum margin known as optimal hyper-plane. The primary idea of SVM is to discover an optimal hyper-plane that categorizes different types of samples [120]. Dimensions of hyper-plane used to rely on features of the dataset. Binary classifications are planned by the standard support vector machines and SVM used a linear separating hyper-plane for binary classification [78, 110]. Generally, SVM has been developed for binary classification. In binary classification, basic support vector machines classifier can work such a way that the kernel function can be pointed out in the input as a high dimensional feature space [22]. SVM is used for text classification, face detection, pattern recognition, hand-written character recognition, etc. SVM has been built for solving the big margin classification problem and, also worked as a statistical learning method based on VC dimensional theory [115]. There are three stages in SVM analysis and those are feature selection, training and testing the classifier, and performance evaluation. SVM is used widely because of its classification accuracy and robustness [14]. SVM performs better with a limited number of samples. There are two types of SVM; those are linear and non-linear SVM [102]. When a dataset is used to get classified into two classes using a straight line, known as linear SVM and this type of SVM is useful for linearly separable data. In linear SVM, problems ranged in their complexity depending on the number of features used [79]. When a dataset can't be classified using a single straight line, known as non-linear SVM and this type of SVM is useful for non-linear data. SVM can process complex data with high accuracy [38]. In SVM, removing one or more support vectors can change the position of the hyper-plane. SVM mainly stands on the idea of structural minimization, which has been concluded by the generalization error that is bounded based on the sum of the training set and a term depending on the Vapnik-Chervonenkis dimension [111]. Generally, SVM consumes more time compared to other machine learning algorithms for training the model for large data sets. The challenge for the SVM tree classifier is how it separates the classes into two separate subsets for the training algorithm [96].

4 Datasets

In terms of executing the classifier, there are two spam datasets used. Those are Spam Corpus and Spambase. The Spam Corpus data set contains 9324 emails; out of which 2387 emails are Spam emails and the rest of 6937 emails are Ham emails. In terms of percentage, 25.60% of emails are Spam and the rest of the 74.40% of emails are Ham. Besides that, the spam corpus dataset contains 500 features or attributes. Spambase data set contains overall 4601 emails, out of the total number of 2788 emails are Spam and the rest of 1813 emails are Ham. In terms of

percentage, 60.59% of emails are spam and the rest of the 39.40% of emails are ham. Except that, the Spambase dataset contains 58 features or attributes. In the following, some of the important features are described (Table 2).

5 Detection approach

This section proposed the detection framework for detecting spam emails using thirteen machine learning classifiers; those are briefly discussed in the previous section. In this research work, two datasets have been used and, both the datasets are having two types of instances. Of which some instances are spam emails and the rest are ham emails instances. Moreover, both datasets are labelled as well as sufficient numbers of instances are also there.

In Fig. 1, the spam email detection framework has been illustrated. In the detection framework, the first step is pre-processing of the dataset. In the pre-processing step, the dataset has been analyzed in terms of attributes and instances. In that analysis, several attributes and instances have been discovered. After that, pre-processed dataset gets ready for performing classification. In the next step, a classifier has been implemented for performing classification tasks. In this step, thirteen different machine learning classifiers have been used one at a time. Separately all the classifiers are executed for both datasets in this step. After performing classification, the dataset has been classified into two separate categories, those are Spam emails and Ham emails.

Multiple datasets have been used in this research work because of measuring the performance of thirteen machine learning classifiers for detecting spam emails in terms of different sizes of the dataset. In the experimental analysis section, the outcome of thirteen machine learning classifiers has been compared in terms of performance evaluation parameters.

Table 2 Useful features of the datasets

Spam Corpus	Spambase
marketing	word_freq_address
credit	word_freq_remove
offer	word_freq_internet
money	word_freq_order
guaranteed	word_freq_mail
dollars	word_freq_receive
insurance	word_freq_people
purchase	word_freq_report
financial	word_freq_free
income	word_freq_email
opportunity	word_freq_credit
shipping	word_freq_money
earn	word_freq_data
debt	word_freq_technology
pay	capital_run_length_average
loans	capital_run_length_longest
spamorlegitimate	capital_run_length_total

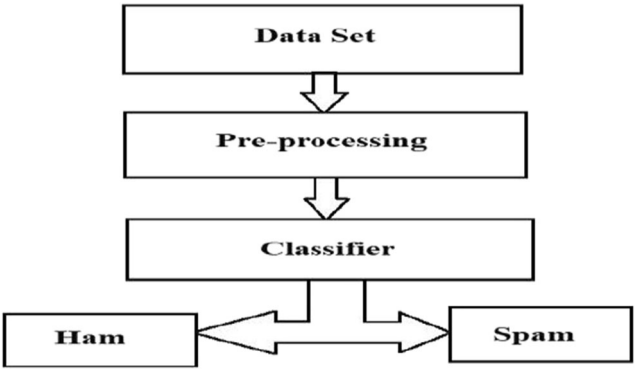

Fig. 1 Illustration of spam emails detecting framework

6 Experimental analysis

6.1 Accuracy

Accuracy is the parameter for measuring the percentage of instances classified correctly.

$$\text{Accuracy} = \frac{\text{Total number of emails classified correctly}}{\text{Total number of emails in the dataset}} \tag{7}$$

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative}} \tag{8}$$

In Fig. 2, the performance of accuracy for all the thirteen machine learning classifiers has been described. In terms of Accuracy, the Random Forest classifier performs better compared to the rest of the classifiers. Random Forest classifier has an accuracy of 99.91% for detecting spam emails from the Spam Corpus dataset. Out of 9324 instances, the Random Forest classifier

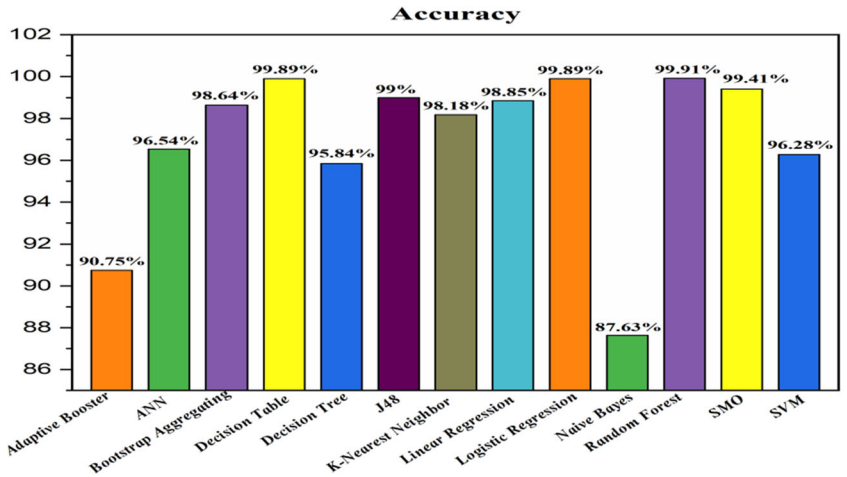

Fig. 2 Accuracy of machine learning classifiers for Spam Corpus dataset

detects 9316 instances correctly. Except for the Random Forest classifier, Decision Table and Logistic Regression classifiers are also perform moderately with 99.89% of accuracy for the Spam Corpus dataset. The Naïve Bayes classifier has the least accuracy compared to the rest of the machine learning classifiers. The Naïve Bayes classifier performs with 87.63% of accuracy for the Spam Corpus dataset. Out of 9324 instances, the Naïve Bayes classifier detects only 8171 instances correctly.

In Fig. 3, the performance of accuracy for all the thirteen machine learning classifiers has been described. In terms of Accuracy, the Random Forest classifier performs better compared to the rest of the classifiers. Random Forest classifier has an accuracy of 99.93% for detecting spam emails from the Spambase dataset. Out of 4601 instances, the Random Forest classifier detects 4598 instances correctly. Except for the Random Forest classifier, J48 and Bootstrap Aggregating classifiers are also performed moderately with 97.17% and 96.72% of accuracy respectively for the Spam Corpus dataset. The Naïve Bayes classifier has the least accuracy compared to the rest of the machine learning classifiers. The Naïve Bayes classifier performs with 79.53% of accuracy for the Spambase dataset. Out of 4601 instances, the Naïve Bayes classifier detects 3659 instances correctly.

6.2 Precision

Precision defines as the percentage of correct spam emails classified from the dataset. Precision also is known as Specificity (true negative rate).

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{9}$$

In Fig. 4, the performance of precision for all the thirteen machine learning classifiers has been described. In terms of precision, Decision Table, Logistic Regression and, Random Forest

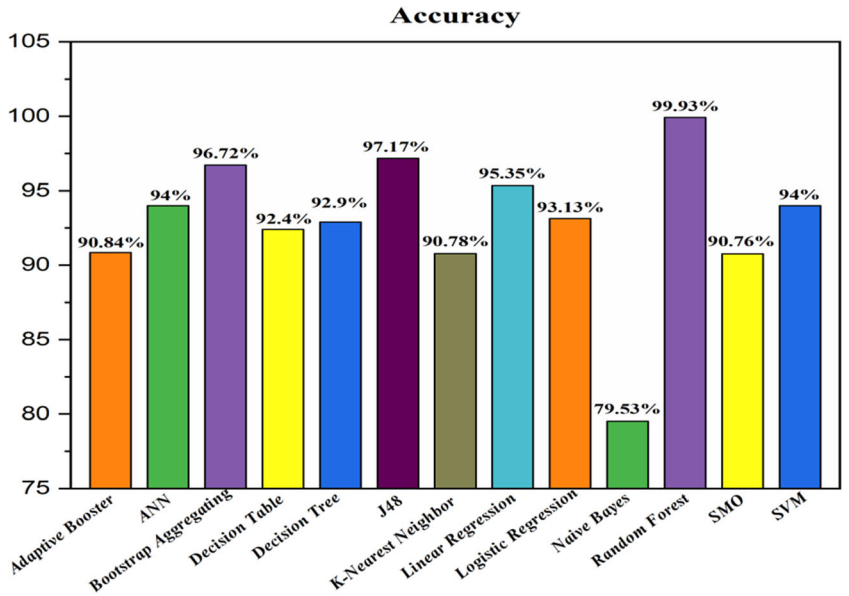

Fig. 3 Accuracy of machine learning classifiers for Spambase dataset

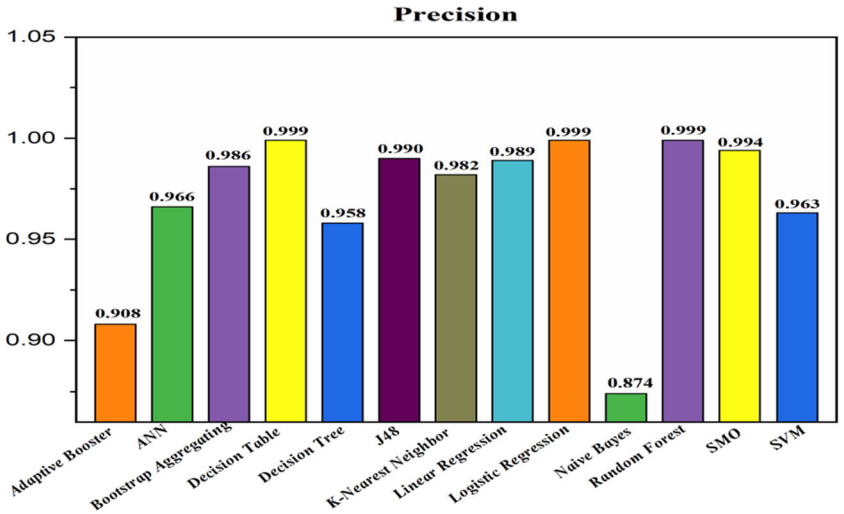

Fig. 4 Precision of machine learning classifiers for Spam Corpus dataset

classifier perform with 0.999 precision for detecting spam emails from the Spam Corpus dataset. The Naïve Bayes classifier has the least precision compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.874 of precision for the Spam Corpus dataset.

In Fig. 5, the performance of precision for all the thirteen machine learning classifiers has been described. In terms of precision, the Random Forest classifier performs with 0.999 precision for detecting spam emails from the Spambase dataset. Except for the Random Forest classifier, J48 and Bootstrap Aggregating classifiers are also performed moderately with 0.972 and 0.967 of precision respectively for the Spambase dataset. The Naïve Bayes classifier has the least precision compared to the rest of the machine learning classifiers. The Naïve Bayes classifier performs with 0.845 precision for the Spambase dataset.

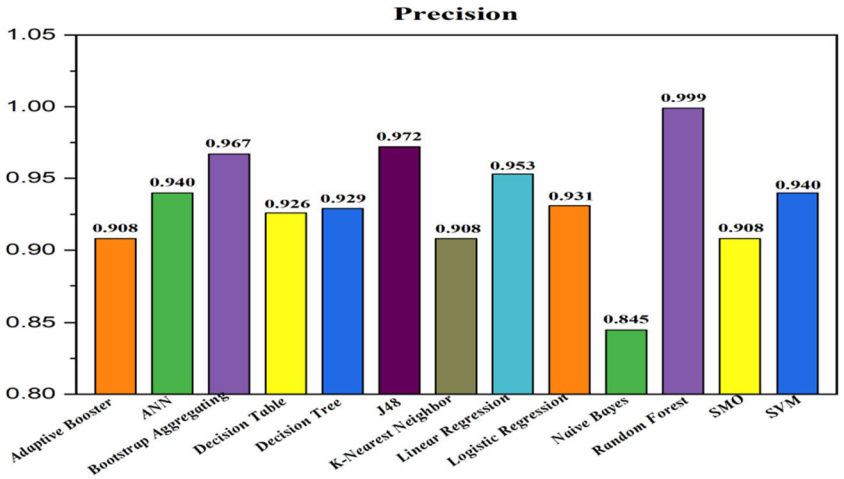

Fig. 5 Precision of machine learning classifiers for Spambase dataset

6.3 Recall

The recall is the parameter for calculating the percentage of spam emails blocked. The recall is also known as Sensitivity (true positive rate or probability of detection).

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{10}$$

In Fig. 6, the performance of recall for all the thirteen machine learning classifiers has been described. In terms of recall, Decision Table, Logistic Regression and, Random Forest classifier performs with 0.999 of recall for detecting spam emails from the Spam Corpus dataset. The Naïve Bayes classifier has the least recall compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.876 of precision for the Spam Corpus dataset.

In Fig. 7, the performance of recall for all the thirteen machine learning classifiers has been described. In terms of recall, the Random Forest classifier performs with 0.999 of recall for detecting spam emails from the Spambase dataset. Except for the Random Forest classifier, J48 and Bootstrap Aggregating classifiers are also performed moderately with 0.972 and 0.967 of recall respectively for the Spambase dataset. The naïve Bayes classifier has the least recall compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.795 of recall for Spambase dataset.

6.4 F-measure

The F-measure defines the average weight-age of Precision and Recall.

$$\text{F}-\text{measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{11}$$

In Fig. 8, the performance of the f-measure for all the thirteen machine learning classifiers has been described. In terms of the f-measure, Decision Table, Logistic Regression and, Random Forest classifier performs with 0.999 of the f-measure for detecting spam emails from the

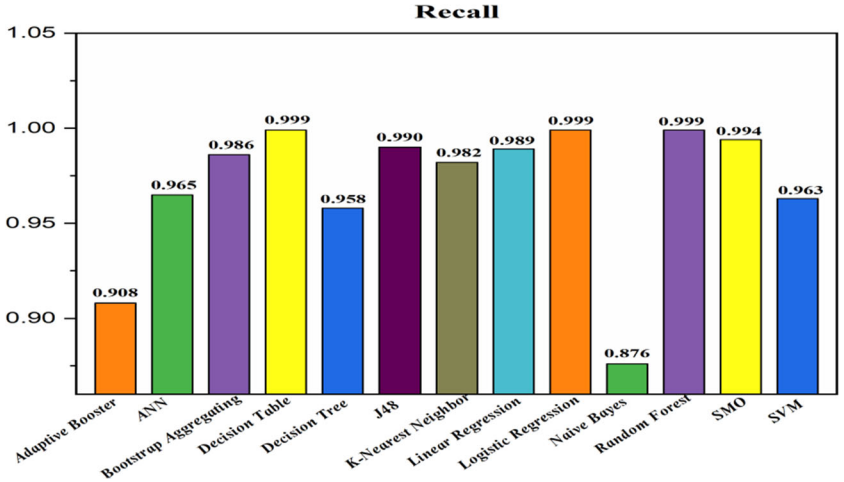

Fig. 6 Recall of machine learning classifiers for Spam Corpus dataset

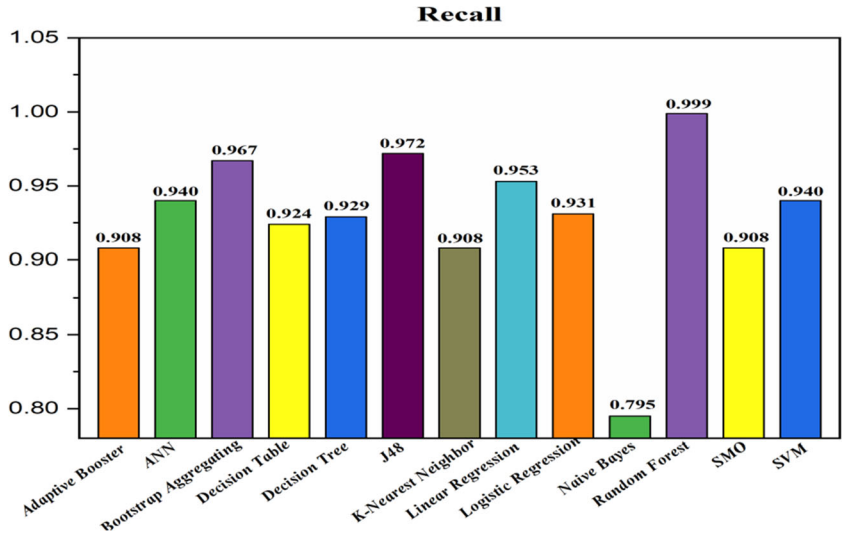

Fig. 7 Recall of machine learning classifiers for Spambase dataset

Spam Corpus dataset. The naïve Bayes classifier has the least f-measure compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.874 of the f-measure for the Spam Corpus dataset.

In Fig. 9, the performance of the f-measure for all the thirteen machine learning classifiers has been described. In terms of the f-measure, the Random Forest classifier performs with 0.999 of the f-measure for detecting spam emails from the Spambase dataset. Except for the Random Forest classifier, J48 and Bootstrap Aggregating classifiers are also performed moderately with 0.972 and 0.967 of the f-measure respectively for the Spambase dataset. The naïve Bayes classifier has the least f-measure compared to the rest of the machine learning classifiers. The Naïve Bayes classifier performs with 0.797 of the f-measure for the Spambase dataset.

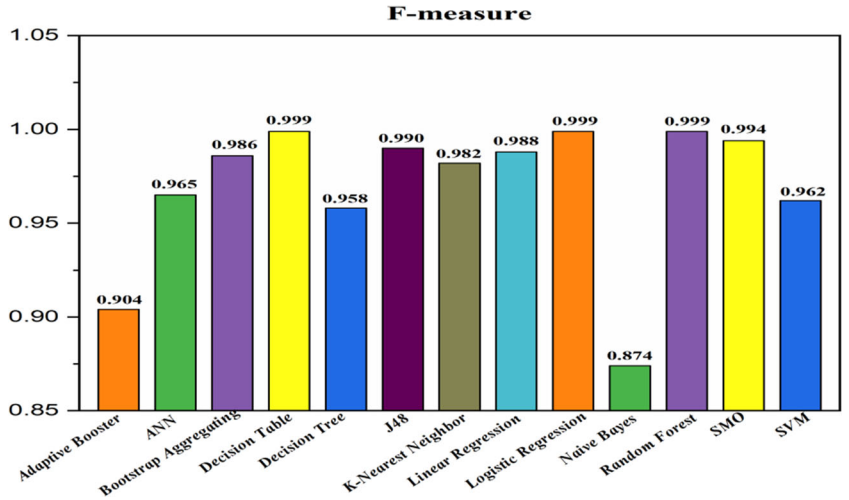

Fig. 8 F-measure of machine learning classifiers for Spam Corpus dataset

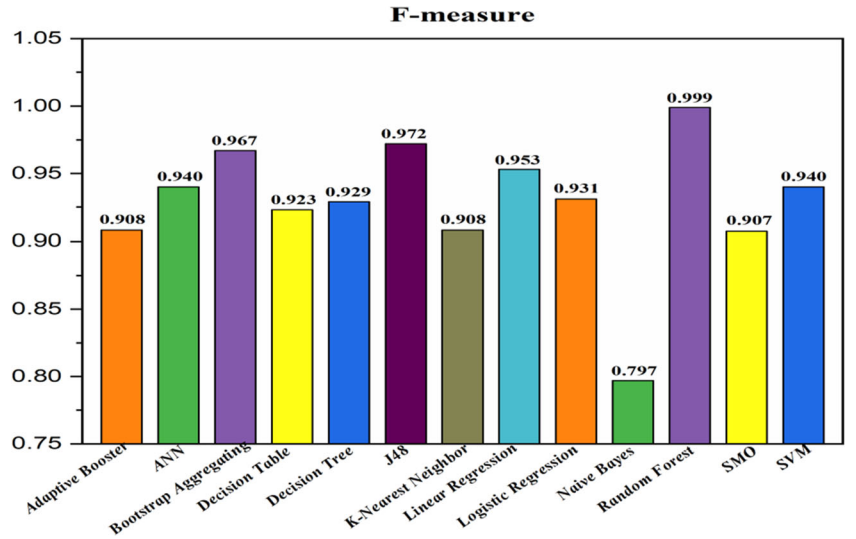

Fig. 9 F-measure of machine learning classifiers for Spambase dataset

6.5 Mathews Correlation Coefficient (MCC)

Mathews Correlation Coefficient is the parameter for measuring the binary classification of two classes.

In Fig. 10, the performance of MCC for all the thirteen machine learning classifiers has been described. In terms of MCC, the Random Forest classifier performs with 0.998 of MCC for detecting spam emails from the Spam Corpus dataset. Except for the Random Forest classifier, Decision Table and Logistic Regression classifiers are also perform moderately with 0.997 of MCC for the Spam Corpus dataset. The Naïve Bayes classifier has the least MCC compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.667 of MCC for Spam Corpus dataset.

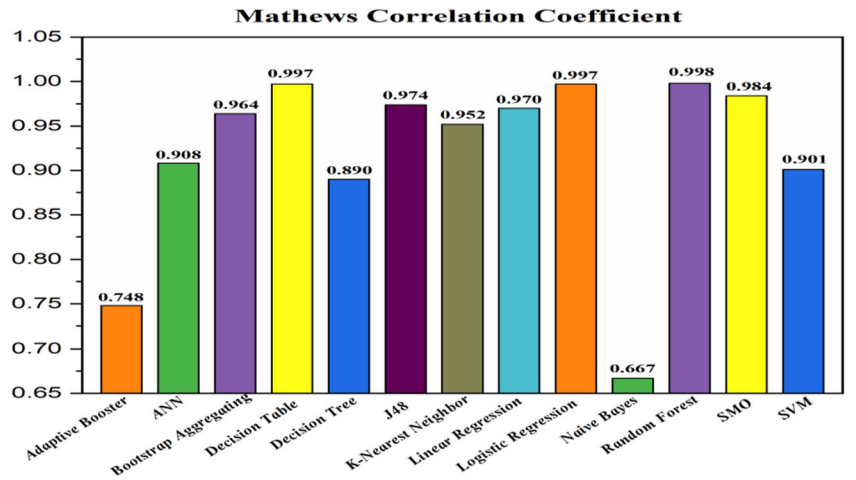

Fig. 10 Mathews Correlation Coefficient of machine learning classifiers for Spam Corpus dataset

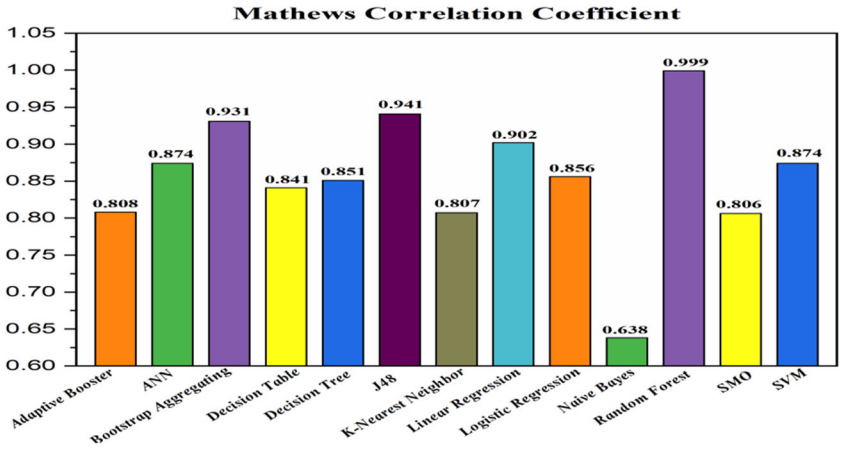

Fig. 11 Mathews Correlation Coefficient of machine learning classifiers for Spambase dataset

In Fig. 11, the performance of MCC for all the thirteen machine learning classifiers has been described. In terms of MCC, the Random Forest classifier performs with 0.999 of MCC for detecting spam emails from the Spambase dataset. Except for the Random Forest classifier, J48 and Bootstrap Aggregating classifiers are also performed moderately with 0.941 and 0.931 of MCC respectively for the Spambase dataset. The Naïve Bayes classifier has the least MCC compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.638 of MCC for Spambase dataset.

6.6 Receiver Operating Characteristic (ROC) Area

Receiver Operating Characteristic area measure the performance of the classifier in a general way.

In Fig. 12, the performance of the ROC area for all the thirteen machine learning classifiers has been described. In terms of the ROC area, Decision Table, Logistic Regression and,

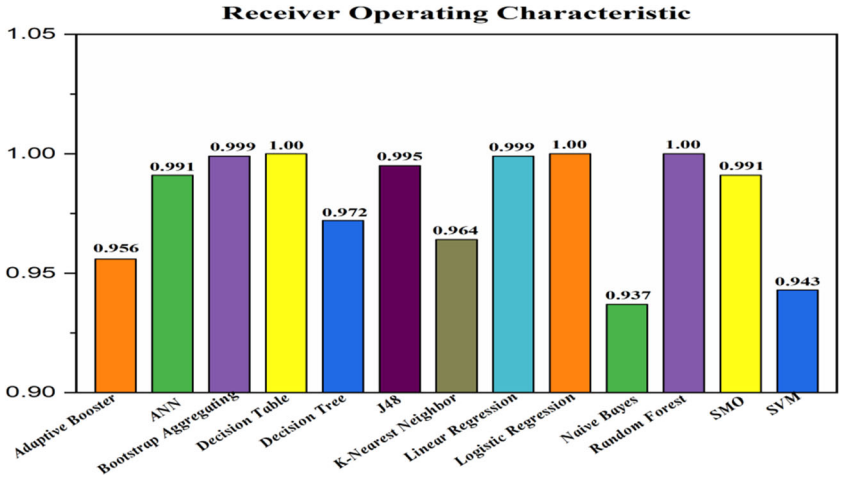

Fig. 12 Receiver Operating Characteristic area of machine learning classifiers for Spam Corpus dataset

Random Forest classifier perform 1.00 of ROC area for detecting spam emails from the Spam Corpus dataset. The naïve Bayes classifier has the least ROC area compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.937 of ROC area for Spam Corpus dataset.

In Fig. 13, the performance of the ROC area for all the thirteen machine learning classifiers has been described. In terms of the ROC area, the Random Forest classifier performs with 1.00 of ROC area for detecting spam emails from the Spambase dataset. Except for the Random Forest classifier, Bootstrap Aggregating and Linear Regression classifiers are also performed moderately with 0.995 and 0.992 of ROC area respectively for the Spambase dataset. The Sequential Minimal Optimization classifier has the least ROC area compared to the rest of the machine learning classifiers. Sequential Minimal Optimization classifier performs with 0.896 of ROC area for Spambase dataset.

6.7 Precision Recall (PRC) Area

The Precision Recall area evaluates the imbalanced dataset in terms of binary classification.

In Fig. 14, the performance of the PRC area for all the thirteen machine learning classifiers has been described. In terms of PRC area, Decision Table, Logistic Regression and, Random Forest classifier perform 1.00 of PRC area for detecting spam emails from the Spam Corpus dataset. The naïve Bayes classifier has the least PRC area compared to the rest of the machine learning classifiers. Naïve Bayes classifier performs with 0.920 of PRC area for Spam Corpus dataset.

In Fig. 15, the performance of the PRC area for all the thirteen machine learning classifiers has been described. In terms of PRC area, the Random Forest classifier performs with 1.00 of PRC area for detecting spam emails from the Spambase dataset. Except for the Random Forest classifier, Bootstrap Aggregating and Linear Regression classifiers are also performed moderately with 0.995 and 0.992 of PRC area respectively for the Spambase dataset. The Sequential Minimal Optimization classifier has the least PRC area compared to the rest of the machine learning classifiers. Sequential Minimal Optimization classifier performs with 0.866 of PRC area for Spambase dataset.

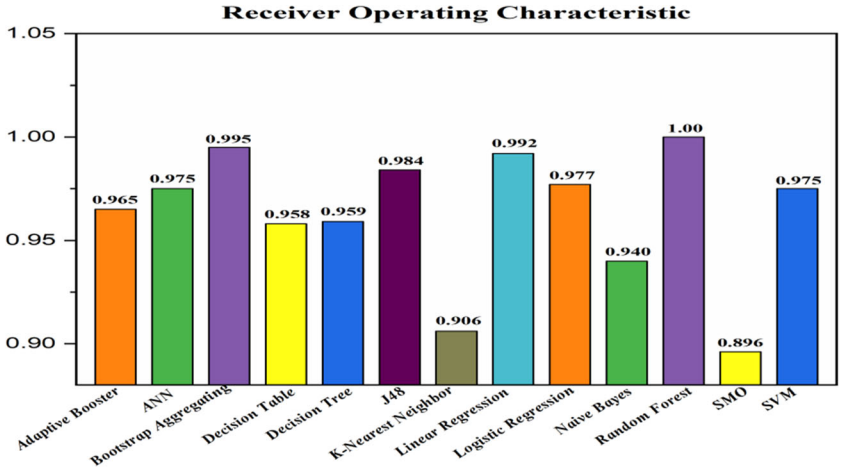

Fig. 13 Receiver Operating Characteristic area of machine learning classifiers for Spambase dataset

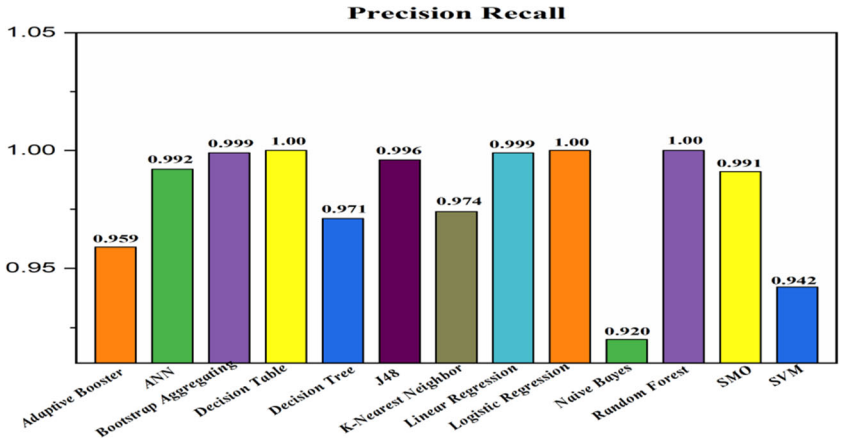

Fig. 14 Precision Recall area of machine learning classifiers for Spam Corpus dataset

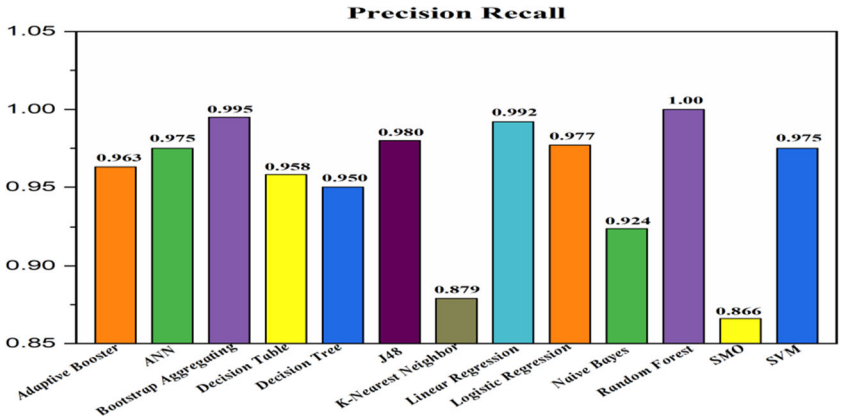

Fig. 15 Precision Recall area of machine learning classifiers for Spambase dataset

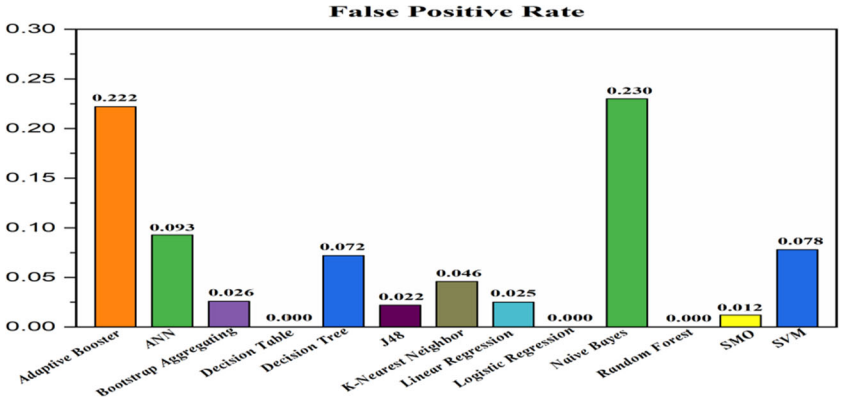

Fig. 16 False Positive Rate of machine learning classifiers for Spam Corpus dataset

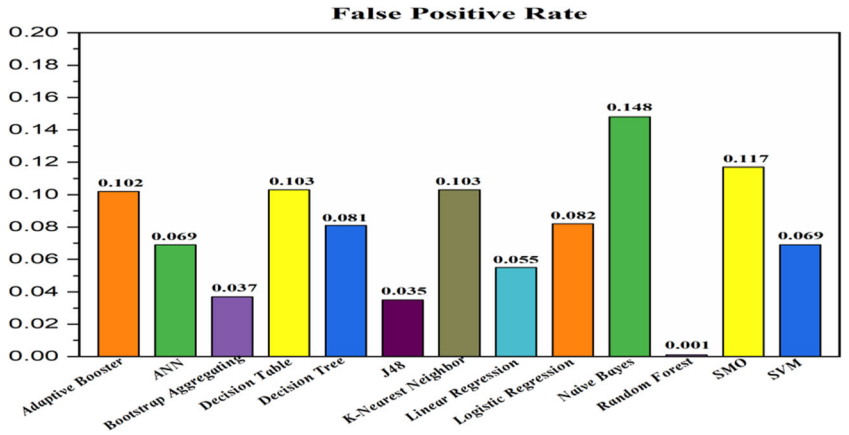

Fig. 17 False Positive Rate of machine learning classifiers for Spambase dataset

6.8 False Positive (FP) Rate

False Positive Rate is the parameter for representing the number of times wrongly predicted by the classifier.

In Fig. 16, the performance of the FP Rate for all the thirteen machine learning classifiers has been described. In terms of FP Rate, Decision Table, Logistic Regression and, Random Forest classifiers are having the least FP Rate compared to the rest of the machine learning classifiers. Decision Table, Logistic Regression and, Random Forest classifiers perform with 0.000 of FP Rate for the Spam Corpus dataset. Naïve Bayes and Adaptive Booster classifiers perform with 0.230 and 0.222 of FP Rate respectively for detecting spam emails from the Spam Corpus dataset.

In Fig. 17, the performance of the FP Rate for all the thirteen machine learning classifiers has been described. In terms of FP Rate, the Random Forest classifier is having the least FP Rate compared to the rest of the machine learning classifiers. Random Forest classifier performs with 0.001 of FP Rate for Spambase dataset. Naïve Bayes and Sequential Minimal classifiers perform with 0.148 and 0.117 of FP Rate respectively for detecting spam emails from the Spambase dataset.

7 Conclusion and future work

In this research work, multiple machine learning classifiers have been implemented for detecting spam emails. The proposed framework has been classifying spam emails and ham emails from the datasets. Two well-known datasets have been used for implementing those thirteen machine learning classifiers. Based on the performance of those thirteen machine learning classifiers experimental analysis has been performed. For experimental analysis, eight parameters have been used. In terms of accuracy, the Random Forest classifier performs better compared to the rest of the machine learning classifiers. Random Forest classifier had the accuracy of 99.91% and 99.93% for the Spam Corpus and Spambase datasets respectively. In terms of accuracy, the Naïve Bayes classifier performs poorly compared to the rest of the machine learning classifiers. The Naïve Bayes classifier had the accuracy of 87.63% and

79.53% for the Spam Corpus and Spambase datasets respectively. In terms of other evaluating parameters also the same result reflected that the Random Forest classifier is the best among all classifiers whereas the Naïve Bayes classifier is the worst among all classifiers.

From the experimental analysis section, it is clear that the Naïve Bayes classifier didn't perform up to the mark for both datasets compared to other machine learning classifiers. In future, planning to improve the performance of the Naïve Bayes classifier based on feature selection for detecting spam emails.

Funding This research work has been written with the financial support of Rashtriya Uchchatar Shiksha Abhiyan (RUSA- Phase 2.0) grant sanctioned vide Letter No. F.24–51/2014-U, Policy (TNMulti-Gen), Dept. of Edn. Govt. of India, Dt. 09.10.2018.

Data availability None.

Declarations

Conflict of interest The authors declare that they have no conflict of interest in the publication of this article.

References

1. Abdulhamid S'i M, Shuaib M, OluwafemiOsho II, Alhassan JK (2018) Comparative Analysis of Classification Algorithms for Email Spam Detection. Int J Comput Netw Inf Secur (IJCNIS) 10(1):60–67. https://doi.org/10.5815/ijcnis.2018.01.07
2. Abiodun OI, Jantan A, Omolara AE, Dada KV, Mohamed NA, Arshad H (2018) "State-of-the-art in artificial neural network applications: A survey", Heliyon, Volume 4, Issue 11, ISSN 2405–8440, https://doi.org/10.1016/j.heliyon.2018.e00938.
3. Ali ABM S, Xiang Y (2007) "Spam Classification Using Adaptive Boosting Algorithm", 6th IEEE/ACIS International Conference on Computer and Information Science (ICIS 2007)
4. Alzahrani A, Rawat DB (2019) "Comparative Study of Machine Learning Algorithms for SMS Spam Detection," 2019 SoutheastCon, pp. 1–6, https://doi.org/10.1109/SoutheastCon42311.2019.9020530.
5. Aminikhanghahi S, Shin S, Wang W, Son SH, Jeon SI (2014) An optimized support vector machine classifier to extract abnormal features from breast microwave tomography data. In Proceedings of the 2014 Conference on research in adaptive and convergent systems (RACS '14). Association for Computing Machinery, New York, NY, pp 111–115. https://doi.org/10.1145/2663761.2664230
6. Anamika, KVL Padmini P, Guduru V, Sangeeta K (2015) Effect of Spam Filter on SPOT Algorithm. In Proceedings of the Third International Symposium on Women in Computing and Informatics (WCI '15). Association for Computing Machinery, New York, NY, USA, 640–643. https://doi.org/10.1145/2791405.2791552
7. Bassiouni M, Ali M, El-Dahshan EA (2018) Ham and spam E-mails classification using machine learning techniques. J Appl Secur Res 13(3):315–331. https://doi.org/10.1080/19361610.2018.1463136
8. Becker BG (1998) "Visualizing decision table classifiers," Proceedings IEEE Symposium on Information Visualization (Cat. No.98TB100258), pp. 102–105, https://doi.org/10.1109/INFVIS.1998.729565.
9. Bedmar IS, Samy D, Martinez JL. (2007) UC3M: classification of semantic relations between nominals using sequential minimal optimization. In Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval '07). Association for Computational Linguistics, USA, 382–385
10. Bertsimas D, Li ML (2020) "Scalable holistic linear regression", Oper Res Lett, https://doi.org/10.1016/j.orl.2020.02.008.
11. David Bienvenido-Huertas, Carlos Rubio-Bellido, Juan Luis Pérez-Ordóñez, Miguel José Oliveira (2020) "Automation and optimization of in-situ assessment of wall thermal transmittance using a random Forest algorithm", Build Environ 168, ISSN 0360-1323, https://doi.org/10.1016/j.buildenv.2019.106479
12. Bin AbdRazak S, Bin Mohamad AF (2013) "Identification of spam email based on information from email header," 2013 13th International Conference on Intellient Systems Design and Applications, Bangi, pp. 347–353

13. Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In proceedings of the fifth annual workshop on computational learning theory (COLT '92). Association for Computing Machinery, New York, NY, USA, pp 144–152. https://doi.org/10.1145/130385.130401
14. Braun AC, Weidner U, Hinz S (2011) "Support vector machines, import vector machines and relevance vector machines for hyperspectral classification — A comparison," 2011 3rd Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), pp. 1–4, https://doi.org/10.1109/WHISPERS.2011.6080861.
15. Breiman, L (1996) Bagging Predict Mach Learn 24, 123–140. https://doi.org/10.1023/A:1018054314350
16. Breiman L (2001) Random For Mach Learn 45:5–32. https://doi.org/10.1023/A:1010933404324
17. Bucurica M, Dogaru R, Dogaru I (2015) "A comparison of Extreme Learning Machine and Support Vector Machine classifiers," 2015 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP), pp. 471–474, https://doi.org/10.1109/ICCP.2015.7312705.
18. Chen M, Challita U, Saad W, Yin C, Debbah M (2019) Artificial Neural Networks-Based Machine Learning for Wireless Networks: A Tutorial. IEEE Commun Surv Tutorials 21(4):3039–3071. https://doi.org/10.1109/COMST.2019.2926625
19. Chharia A, Gupta RK (2013) "Email classifier: An ensemble using probability and rules," 2013 Sixth International Conference on Contemporary Computing (IC3), pp. 130–136, https://doi.org/10.1109/IC3.2013.6612176.
20. Cho J, Kim S (2020) "Personal and social predictors of use and non-use of fitness/diet app: application of random Forest algorithm", Telematics Inf 55, ISSN 0736-5853, https://doi.org/10.1016/j.tele.2019.101301
21. Cristianni N, Shawe-Talyor J (2000) "An introduction to support vector machines", Cambridge Uninversity Press
22. Cui J, Wang Y (2011) A novel approach of analog circuit fault diagnosis using support vector machines classifier. Measurement 44(1):281–289, ISSN 0263-2241. https://doi.org/10.1016/j.measurement.2010.10.004
23. Diale M, Van Der Walt C, Celik T, Modupe A (2016) "Feature selection and support vector machine hyper-parameter optimisation for spam detection," 2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), pp. 1–7, https://doi.org/10.1109/RoboMech.2016.7813162.
24. Digamberrao KS, Prasad RS (2018) Author Identification using Sequential Minimal Optimization with rule-based Decision Tree on Indian Literature in Marathi. Proced Comput Sci 132:1086–1101, ISSN 1877–0509. https://doi.org/10.1016/j.procs.2018.05.024
25. Dong Y, Ma X, Fu T (2020) Electrical load forecasting: A deep learning approach based on K-nearest neighbors, Appl Soft Comput J, https://doi.org/10.1016/j.asoc.2020.106900.
26. Dudani SA (1976) The distance-weighted k-nearest-neighbor rule. Trans Syst Man Cybern SMC-6(4): 325–327. https://doi.org/10.1109/TSMC.1976.5408784
27. Edla DR, Mangalorekar K, Dhavalikar G, Dodia S (2018) Classification of EEG data for human mental state analysis using Random Forest Classifier. Procedia Comput Sci 132:1523–1532, ISSN 1877–0509. https://doi.org/10.1016/j.procs.2018.05.116
28. Ernawati S, Yulia ER, Frieyadie, Samudi (2018) "Implementation of The Naïve Bayes Algorithm with Feature Selection using Genetic Algorithm for Sentiment Review Analysis of Fashion Online Companies," 2018 6th International Conference on Cyber and IT Service Management (CITSM), pp. 1–5, https://doi.org/10.1109/CITSM.2018.8674286.
29. Fujiwara Y, Ida Y, Kanai S, Kumagai A, Arai J, Ueda N (2019) Fast random Forest algorithm via incremental upper bound. In proceedings of the 28th ACM international conference on information and knowledge management (CIKM '19). Association for Computing Machinery, New York, NY, USA, pp 2205–2208. https://doi.org/10.1145/3357384.3358092
30. Garner SR (n.d.) "WEKA: The Waikato Environment for Knowledge Analysis", Available: https://www.cs.waikato.ac.nz/~ml/publications/1995/Garner95-WEKA.pdf
31. Gavankar SS, Sawarkar SD (2017) "Eager decision tree," 2017 2nd International Conference for Convergence in Technology (I2CT), pp. 837–840, https://doi.org/10.1109/I2CT.2017.8226246.
32. Gbenga DE, Christopher N, Yetunde DC (2017) Performance Comparison of Machine Learning Techniques for Breast Cancer Detection. Nova J Eng Appl Sci 6(1):1–8. https://doi.org/10.20286/nova-jeas-060105
33. Gomes SR et al. (2017) "A comparative approach to email classification using Naive Bayes classifier and hidden Markov model," 2017 4th International Conference on Advances in Electrical Engineering (ICAEE), Dhaka, pp. 482–487
34. Gong C, Zhi-gang S, Wang P-h, Wang Q, Yang Y (2021) Evidential instance selection for K-nearest neighbor classification of big data. Int. J. Approx. Reason. 138:123–144, ISSN 0888-613X. https://doi.org/10.1016/j.ijar.2021.08.006

35. Guo Y., Bai L., Lao S., Wu S., Lew M.S. (2014) A Comparison between Artificial Neural Network and Cascade-Correlation Neural Network in Concept Classification. In: Ooi W.T., Snoek C.G.M., Tan H.K., Ho CK., Huet B., Ngo CW. (eds) Advances in Multimedia Information Processing – PCM 2014. PCM 2014. Lecture notes in computer science, vol 8879. Springer, Cham https://doi.org/10.1007/978-3-319-13168-9_26
36. Gupta P, Dubey RK, Mishra S (2019) "Detecting Spam Emails/Sms Using Naive Bayes and Support Vector Machine", Int J Sci Technol Res, Volume 8, Issue 11
37. Hassan MA, Mtetwa N (2018) "Feature Extraction and Classification of Spam Emails," 2018 5th international conference on Soft Computing & Machine Intelligence (ISCMI), Nairobi, Kenya, pp. 93–98
38. He L, Yang X, Lu H (2007) "A Comparison of Support Vector Machines Ensemble for Classification," 2007 International Conference on Machine Learning and Cybernetics, pp. 3613–3617, https://doi.org/10.1109/ICMLC.2007.4370773.
39. Heredia B, Khoshgoftaar TM, Prusa J, Crawford M (2016) "An Investigation of Ensemble Techniques for Detection of Spam Reviews," 2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 127–133, https://doi.org/10.1109/ICMLA.2016.0029.
40. Huang X, Shi L, Suykens JAK (2015) Sequential minimal optimization for SVM with pinball loss. Neurocomputing 149(Part C):1596–1603, ISSN 0925-2312. https://doi.org/10.1016/j.neucom.2014.08.033
41. Jain V, Phophalia A (2019) "Exponential Weighted Random Forest for Hyperspectral Image Classification," IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium, pp. 3297–3300, https://doi.org/10.1109/IGARSS.2019.8897862.
42. Jiang L, Cai Z, Wang D, Jiang S (2007) "Survey of Improving K-Nearest-Neighbor for Classification," Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007), pp. 679–683, https://doi.org/10.1109/FSKD.2007.552.
43. Joshi AV (2020) Decision trees. In: Machine Learning and Artificial Intelligence. Springer, Cham. https://doi.org/10.1007/978-3-030-26622-6_6
44. Kadlček F, Fučík O (2013) "Fast and energy efficient AdaBoost classifier", in proceedings of the 10th FPGAworld conference (FPGAworld '13). Assoc Comput Mach New York NY USA 2:1–5. https://doi.org/10.1145/2513683.2513685
45. Kalbhor M, Shrivastava S, Ujjainiya B (2013) "An artificial immune system with local feature selection classifier for spam filtering," in 2013 fourth international conference on computing, communications and networking technologies (ICCCNT), Tiruchengode, India pp. 1–7.https://doi.org/10.1109/ICCCNT.2013.6726691
46. Kalmegh SR (2018) Comparative analysis of the WEKA classifiers rules Conjunctiverule&Decisiontable on Indian news dataset by using different test mode. Int J Eng Sci Invent (IJESI) 7(2):01–09
47. Kang K, Gao F, Feng J (2018) "A New Multi-Layer Classification Method Based on Logistic Regression," 2018 13th International Conference on Computer Science & Education (ICCSE), pp. 1–4, https://doi.org/10.1109/ICCSE.2018.8468725
48. Kaur J, Baghla S (2017) Modified decision table classifier by using decision support and confidence in online shopping dataset. Int J Comput Eng Technol 8(6):83–88
49. Kohavi R (1995) The power of decision tables. In proceedings of the 8th European conference on machine learning (ECML'95). Springer-Verlag, Berlin, Heidelberg, pp 174–189. https://doi.org/10.1007/3-540-59286-5_57
50. Kohavi R, Sommerfield D (1998) Targeting business users with decision table classifiers. In proceedings of the fourth international conference on knowledge discovery and data mining (KDD'98). AAAI press, 249–253
51. Kontsewaya Y, Antonov E, Artamonov A (2021) Evaluating the effectiveness of machine learning methods for spam detection. Proced Comput Sci 190:479–486, ISSN 1877-0509. https://doi.org/10.1016/j.procs.2021.06.056
52. Kramer O (2013) K-nearest neighbors. In: dimensionality reduction with unsupervised nearest neighbors. Intelligent systems reference library, vol 51. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-38652-7_2
53. Lei H, Long Q (2011) "Locate potential Support Vectors for faster Sequential Minimal Optimization," 2011 Seventh International Conference on Natural Computation, pp. 367–372, https://doi.org/10.1109/ICNC.2011.6022107.
54. Li J, Huang S, He R, Qian K (2008) "Image Classification Based on Fuzzy Support Vector Machine," 2008 International Symposium on Computational Intelligence and Design, pp. 68–71, https://doi.org/10.1109/ISCID.2008.51.
55. Lin C-F, Wang S-D (2002) Fuzzy support vector machines. Trans Neural Netw 13(2):464–471. https://doi.org/10.1109/72.991432

56. Lin Z, Qiu D, Ergu D, Ying C, Liu K (2019) "A study on predicting loan default based on the random forest algorithm", Proced Comput Sci, Volume 162, Pages 503–513, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2019.12.017.
57. Lin L, Dekkers IA, Tao Q, Lamb HJ (n.d.) "Novel artificial neural network and linear regression based equation for estimating visceral adipose tissue volume, Clin Nutr", https://doi.org/10.1016/j.clnu.2020.02.013.
58. Liu Y-Z, Yao H-X, Gao W, Zhao D-B (2005) Single sequential minimal optimization: an improved SVMs training algorithm. 2005 Int Conf Mach Learn Cybern 7:4360–4364. https://doi.org/10.1109/ICMLC.2005.1527705
59. Lv C, Chen D-R (2018) "Interpretable Functional Logistic Regression", CSAE '18, October 22–24, Hohhot, China, https://doi.org/10.1145/3207677.3277962
60. Ma CJ, Ding ZS (2020) "Improvement of k-nearest neighbor algorithm based on double filtering," 2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE), pp. 1567–1570, https://doi.org/10.1109/ICMCCE51767.2020.00343.
61. Matharasi B, Senthilrajan A (2017) Sentiment Analysis of Twitter Data using Naive bayes with Unigran Approach. Int J Sci Res Publ 7(Issue – 5) ISSN: 2250-3153:337–341
62. Mendez JR, Cotos-Yanez TR, Ruano-Ordas D (2019) A new semantic-based feature selection method for spam filtering. Appl Soft Comput J 76:89–104
63. Moon S-H, Kim Y-H (2019) "An improved forecast of precipitation type using correlation-based feature selection and multinomial logistic regression", Atmos Res, https://doi.org/10.1016/j.atmosres.2020.104928
64. More AS, Rana DP (2017) "Review of random forest classification techniques to resolve data imbalance," 2017 1st International Conference on Intelligent Systems and Information Management (ICISIM), pp. 72–78, https://doi.org/10.1109/ICISIM.2017.8122151.
65. Nasreen M Shajideen, BV (2018) "Spam filtering: a comparison between different machine learning classifiers", proceedings of the 2nd international conference on electronics, communication and aerospace technology (ICECA 2018)
66. Noronha DH, Torquato MF, Fernandes MAC (2019) A parallel implementation of sequential minimal optimization on FPGA. Microprocess Microsyst 69:138–151, ISSN 0141-9331. https://doi.org/10.1016/j.micpro.2019.06.007
67. Okfalisa, IG, Mustakim, Reza NGI (2017) "Comparative analysis of k-nearest neighbor and modified k-nearest neighbor algorithm for data classification," 2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE), pp. 294–298, https://doi.org/10.1109/ICITISEE.2017.8285514.
68. Osegi EN, Jumbo EF (2021) "Comparative analysis of credit card fraud detection in simulated annealing trained artificial neural network and hierarchical temporal memory", Mach Learn Appl 6, ISSN 2666-8270, https://doi.org/10.1016/j.mlwa.2021.100080
69. Paing MP, Pintavirooj C, Tungjitkusolmun S, Choomchuay S, Hamamoto K (2018) "Comparison of Sampling Methods for Imbalanced Data Classification in Random Forest," 2018 11th Biomedical Engineering International Conference (BMEiCON), pp. 1–5, https://doi.org/10.1109/BMEiCON.2018.8609946.
70. Panhalkar AR, Doye DD (2021) "Optimization of decision trees using modified African buffalo algorithm", J King Saud Univ Comput Inf Sci, ISSN 1319-1578, https://doi.org/10.1016/j.jksuci.2021.01.011.
71. Panigrahi PK (2012) "A Comparative Study of Supervised Machine Learning Techniques for Spam E-mail Filtering," 2012 Fourth International Conference on Computational Intelligence and Communication Networks, pp. 506–512, https://doi.org/10.1109/CICN.2012.14.
72. Panigrahi R, Borah S (2018) Rank Allocation to J48 Group of Decision Tree Classifiers using Binary and Multiclass Intrusion Detection Datasets. Int Conf Comput Intell Data Sci (ICCIDS 2018), Procedia Comput Therm Sci 132:323–332
73. Patel DR, Kiran MB (n.d.) "A non-contact approach for surface roughness prediction in CNC turning using a linear regression model", Mater Today Proceed, https://doi.org/10.1016/j.matpr.2019.12.029
74. Patel R, Thakkar P (2014) "Opinion Spam Detection Using Feature Selection," 2014 International Conference on Computational Intelligence and Communication Networks, pp. 560–564, https://doi.org/10.1109/CICN.2014.127.
75. Patil S, Kulkarni U (2019) "Accuracy Prediction for Distributed Decision Tree using Machine Learning approach," 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), pp. 1365–1371, https://doi.org/10.1109/ICOEI.2019.8862580.
76. Paul A, Mukherjee DP (2016) Reinforced random forest. In proceedings of the tenth Indian conference on computer vision, graphics and image processing (ICVGIP '16). Assoc Comput Mach New York NY USA 1:1–8. https://doi.org/10.1145/3009977.3010003

77. Pelle R, Alcântara C, Moreira VP (2018) “A Classifier Ensemble for Offensive Text Detection”, WebMedia ‘18, October 16–19, Salvador-BA, Brazil, https://doi.org/10.1145/3243082.3243111
78. Peng X, Xu D (2013) A twin-hypersphere support vector machine classifier and the fast learning algorithm. Inf. Sci. 221:12–27, ISSN 0020-0255. https://doi.org/10.1016/j.ins.2012.09.009
79. Pisner DA, Schnyer DM (2020) “Chapter 6 - Support vector machine”, Editor(s): Andrea Mechelli, Sandra Vieira, Machine Learning, Academic Press, Pages 101–121, ISBN 9780128157398, https://doi.org/10.1016/B978-0-12-815739-8.00006-7.
80. Platt, J (1998) Sequential minimal optimization : a fast algorithm for training support vector machines. Microsoft Res Tech Rep
81. Platt JC (1999) Fast training of support vector machines using sequential minimal optimization. Advances in kernel methods: support vector learning. MIT press, Cambridge, MA, USA, pp 185–208
82. Platt JC (1999) Using analytic QP and sparseness to speed training of support vector machines. In proceedings of the 1998 conference on advances in neural information processing systems II. MIT press, Cambridge, MA, USA, pp 557–563
83. Provost, J (1999) “Na ive-Bayes vs. Rule-Learning in Classification of Email.”Available: http://www.cs.utexas.edu/ftp/AI-Lab/tech-reports/UT-AI-TR-99-284.pdf
84. Quinlan JR (1996) Learning decision tree classifiers. ACM Comput Surv 28(1):71–72. https://doi.org/10.1145/234313.234346
85. Rachida I, Abdelwahed N, Sanaa E F (2019) “J48 Algorithms of machine learning for predicting user’s the acceptance of an E-orientation Systems”, SCA2019, October 2–4, Casablanca, Morocco
86. Rathod SB, Pattewar TM (2015) "Content based spam detection in email using Bayesian classifier," 2015 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, pp. 1257–1261, https://doi.org/10.1109/ICCSP.2015.7322709.
87. Sahingoz OK, Buber E, Onder Demir BD (2019) Machine learning based phishing detection from URLs. Exp Syst Appl 117:345–357, ISSN 0957–4174. https://doi.org/10.1016/j.eswa.2018.09.029
88. Sahoo SR, Gupta BB (2020) Classification of spammer and nonspammer content in online social network using genetic algorithm-based feature selection. Enterp Inf Syst 14(5):710–736. https://doi.org/10.1080/17517575.2020.1712742
89. Saidani N, Adi K, Allili MS (2020) “A semantic-based classification approach for an enhanced spam detection”, Comput Secur 94, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2020.101716
90. Schapire RE (2013) Explaining AdaBoost. In: Schölkopf B, Luo Z, Vovk V (eds) Empirical Inference. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-41136-6_5
91. ShafighAski A, Sourati NK (2016) Proposed efficient algorithm to filter spam using machine learning techniques. Pac Sci Rev A Nat Sci Eng 18(2):145–149
92. Shah N, Jain S (2019) "Detection of Disease in Cotton Leaf using Artificial Neural Network," 2019 Amity International Conference on Artificial Intelligence (AICAI), pp. 473–476, https://doi.org/10.1109/AICAI.2019.8701311.
93. Shahbudin S, Hussain A, Samad SA, Md Tahir N (2008) "Training and analysis of Support Vector Machine using Sequential Minimal Optimization," 2008 IEEE International Conference on Systems, Man and Cybernetics, pp. 373–378, https://doi.org/10.1109/ICSMC.2008.4811304.
94. Sharma A, Suryawanshi A (2016) “A Novel Method for Detecting Spam Email using KNN Classification with Spearman Correlation as Distance Measure”, Int J Comput Appl (0975–8887) Volume 136, No.6, https://doi.org/10.5120/ijca2016908471
95. Sharma AK, Yadav R (2015) “Spam Mails Filtering Using Different Classifiers with Feature Selection and Reduction Techniques”, 2015 Fifth International Conference on Communication Systems and Network Technologies
96. Shubhangi DC, Hiremath PS (2009) Support vector machine (SVM) classifier for brain tumor detection. In proceedings of the international conference on advances in computing, communication and control (ICAC3 '09). Association for Computing Machinery, New York, NY, USA, pp 444–448. https://doi.org/10.1145/1523103.1523191
97. Singh AK, Bhushan S, Vij S(2019) “filtering spam messages and mails using fuzzy C means algorithm”, 2019 4th international conference on internet of things: smart innovation and usages (IoT-SIU), Ghaziabad, India, pp. 1–5
98. Subasi A, Alzahrani S, Aljuhani A, Aljedani M (2018) "Comparison of Decision Tree Algorithms for Spam E-mail Filtering," 2018 1st international conference on computer applications & information security (ICCAIS), Riyadh, pp. 1–5
99. Sun S, Huang R (2010) "An adaptive k-nearest neighbor algorithm," 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery, pp. 91–94, https://doi.org/10.1109/FSKD.2010.5569740.

100. Suriya Prakash J, Annamalai Vignesh K, Ashok C, Adithyan R (2012) "Multi class Support Vector Machines classifier for machine vision application," 2012 International Conference on Machine Vision and Image Processing (MVIP), pp. 197–199, https://doi.org/10.1109/MVIP.2012.6428794.
101. Susai Mary J, Sai Balaji MA, Krishnakumari A, Nakandhrakumar RS, Dinakaran D (2019) Monitoring of drill runout using Least Square support vector machine classifier. Measurement 146:24–34, ISSN 0263-2241. https://doi.org/10.1016/j.measurement.2019.05.102
102. Suthaharan S (2016) Support vector machine. In: machine learning models and algorithms for big data classification. Integrated series in information systems, vol 36. Springer, Boston, MA. https://doi.org/10.1007/978-1-4899-7641-3_9
103. Tina R Patil, SSS (2013) "Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification", Int J Comput Sci Appl Vol. 6, No.2 ISSN: 0974–1011
104. Tran T, Tsai P, Jan T (2008) "An adjustable combination of linear regression and modified probabilistic neural network for anti-spam filtering," in ICPR 2008 19th international conference on pattern recognition, Tampa, FLhttps://doi.org/10.1109/ICPR.2008.4761358
105. Tretyakov K (n.d.) "Machine Learning Techniques in Spam Filtering", Available: https://courses.cs.ut.ee/2004/dm-seminar-spring/uploads/Main/P06.pdf
106. Tseng C, Chen M (2009) "Incremental SVM Model for Spam Detection on Dynamic Email Social Networks," 2009 International Conference on Computational Science and Engineering, pp. 128–135, https://doi.org/10.1109/CSE.2009.260.
107. Turčaník M (2015) "Packet filtering by artificial neural network," Int Conf Mil Technol (ICMT), 2015, pp. 1–4, https://doi.org/10.1109/MILTECHS.2015.7153739.
108. Urmaliya A, Singhai J (2013) "Sequential minimal optimization for support vector machine with feature selection in breast cancer diagnosis," 2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013), pp. 481–486, https://doi.org/10.1109/ICIIP.2013.6707638.
109. Vanhoenshoven F, Nápoles G, Falcon R, Vanhoof K, Köppen M (2016) "Detecting malicious URLs using machine learning techniques," 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, pp. 1–8, https://doi.org/10.1109/SSCI.2016.7850079.
110. Vapnik VN (1998) Statistical learning theory. John Wiley & Sons
111. Vapnik VN (1999) An overview of statistical learning theory. Trans Neural Netw 10(5):988–999. https://doi.org/10.1109/72.788640
112. Vapnik VN (2000) Methods of pattern recognition. In: The Nature of Statistical Learning Theory. Statistics for Engineering and Information Science. Springer, New York, NY. https://doi.org/10.1007/978-1-4757-3264-1_6
113. Vijayanand R, Devaraj D, Kannapiran B (2018) Intrusion detection system for wireless mesh network using multiple support vector machine classifiers with genetic-algorithm-based feature selection. Comput Secur 77:304–314, ISSN 0167-4048. https://doi.org/10.1016/j.cose.2018.04.010
114. Wang SC (2003) Artificial neural network. In: interdisciplinary computing in Java programming. The springer international series in engineering and computer science, vol 743. Springer, Boston, MA. https://doi.org/10.1007/978-1-4615-0377-4_5
115. Wang L-S, Xu Y-T, Zhao L-S (2005) "A kind of hybrid classification algorithm based on rough set and support vector machine," 2005 International Conference on Machine Learning and Cybernetics, pp. 1676-1679 Vol. 3, https://doi.org/10.1109/ICMLC.2005.1527214.
116. Wang S, Aggarwal C, Liu H (2018) Random-Forest-Inspired Neural Networks. ACM Trans Intell Syst Technol 9(6):Article 69. https://doi.org/10.1145/3232230
117. Wang F, Wang Q, Nie F, Li Z, Yu W, Ren F (2020, ISSN 0031-3203) A linear multivariate binary decision tree classifier based on K-means splitting. Pattern Recognit 107:107521. https://doi.org/10.1016/j.patcog.2020.107521
118. Wei R, Ghosal S (2020) Contraction properties of shrinkage priors in logistic regression. J Stat Plann Infer 207:215–229. https://doi.org/10.1016/j.jspi.2019.12.004
119. Witt G (2012) Chapter 3 - A brief history of rules, Editor(s): Graham Witt, Writing Effective Business Rules, Morgan Kaufmann, Pages 25–63, ISBN 9780123850515, https://doi.org/10.1016/B978-0-12-385051-5.00003-3.
120. Wu S, Tong X, Wang W, Xin G, Wang B, Zhou Q (2018) Website defacements detection based on support vector machine classification method. In proceedings of the 2018 international conference on computing and data engineering (ICCDE 2018). Association for Computing Machinery, New York, NY, USA, pp 62–66. https://doi.org/10.1145/3219788.3219804
121. Yang F (2019) "An Extended Idea about Decision Trees," 2019 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 349–354, https://doi.org/10.1109/CSCI49370.2019.00068.

122. Yasin W, Ibrahim H Intelligent Cooperative Least Recently Used Web Caching Policy based on J48 Classifier. iiWAS2014 Hanoi, Vietnam
123. Yuan P, Ren S, Xu H, Chen J (2018) "Chrysanthemum Abnormal Petal Type Classification using Random Forest and Over-sampling," 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 275–278, https://doi.org/10.1109/BIBM.2018.8621234.
124. Zhang Z (2018) Artificial neural network. In: Multivariate Time Series Analysis in Climate and Environmental Research. Springer, Cham. https://doi.org/10.1007/978-3-319-67340-0_1