International Journal of Advanced Technology in Engineering and Science
Vol. No. 11, Issue No. 04, April 2023
www.ijates.com
ijates
ISSN 2348 - 7550

# Email Spam Detection Using Machine Learning Algorithms

**D. Chandra Hasini Lakshmi ,J. Mahalakshmi ,G. Sujatha , A. Ravi**

*( Under the guidance of **Dr. A. Balaji**, Professor, Head of the Department,*
*Department of Computer Science and Engineering, Tirumala Engineering College)*

**ABSTRACT**

*Email Spam has become a major problem nowadays, with Rapid growth of internet users, Email spams is also increasing. People are using them for illegal and unethical conducts, phishing and fraud. Sending malicious link through spam emails which can harm our system and can also seek in into your system. Creating a fake profile and email account is much easy for the spammers, they pretend like a genuine person in their spam emails, these spammers target those peoples who are not aware about these frauds. So, it is needed to Identify those spam mails which are fraud, this project will identify those spam by using techniques of machine learning, this paper will discuss the machine learning algorithms and apply all these algorithm on our data sets and best algorithm is selected for the email spam detection having best precision and accuracy.*

***Keywords: Machine learning, Naïve Bayes, support vector machine-nearest neighbour, random forest, bagging, boosting, neural networks***.

## I. INTRODUCTION

Email or electronic mail spam refers to the "using of email to send unsolicited emails or advertising emails to a group of recipients. Unsolicited emails mean the recipient has not granted permission for receiving those emails. "The popularity of using spam emails is increasing since last decade. Spam has become a big misfortune on the internet.

Spam is a waste of storage, time and message speed. Automatic email filtering may be the most effective method of detecting spam but nowadays spammers can easily bypass all these spam filtering applications easily. Several years ago, most of the spam can be blocked manually coming from certain email addresses. Machine learning approach will be used for spam detection. Major approaches adopted closer to junk mail filtering encompass "text analysis, white and blacklists of domain names, and community-primarily based techniques". Text assessment of contents of mails is an extensively used method to the spams. Many answers deployable on server and purchaser aspects are available. Naive Bayes is one of the utmost well-known algorithms applied in these procedures. However, rejecting sends essentially dependent on content examination can be a difficult issue in the event of bogus positives. Regularly clients and organizations would not need any legitimate messages to be lost. The boycott approach has been probably the soonest technique pursued for the separating of spams. The

technique is to acknowledge all the sends other than those from the area/electronic mail ids. Expressly boycotted. With more up to date areas coming into the classification of spamming space names this technique keeps an eye on no longer work so well. The white list approach is the approach of accepting the mails from the domain names/addresses openly whitelisted and place others in a much less importance queue, that is delivered most effectively after the sender responds to an affirmation request sent through the "junk mail filtering system". Spam and Ham: According to Wikipedia "the use of electronic messaging systems to send unsolicited bulk messages, especially mass advertisement, malicious links etc." are called as spam. "Unsolicited means that those things which you didn't asked for messages from the sources. So, if you do not know about the sender the mail can be spam. People generally don't realize they just signed in for those mailers when they download any free services, software or while updating the software. "Ham" this term was given by Spam Bayes around 2001 and it is defined as "Emails that are not generally desired and is not considered spam"

Machine learning approaches are more efficient, a set of training data is used, these samples are the set of email which are pre classified. Machine learning approaches have a lot of algorithms that can be used for email filtering. These algorithms include "Naïve Bayes, support vector machines, Neural Networks, K-nearest neighbor, Random Forests etc."

## II. LITERATURE SURVEY

There is some related work that apply machine learning methods in email spam detection, A. Karim, S. Azam, B. Shanmugam, K. Kannoorpatti and M. Alazab.[ii] They describe a focused literature survey of Artificial Intelligence Revised (AI) and Machine learning methods for email spam detection. K. Agarwal [3] and T. Kumar. Harisinghaney et al. (2014) [4]and Mohamad & Selamat (2015) [v] have used the "image and textual dataset for the e-mail spam detection with the use of various methods. Harisinghaney et al. (2014) [iv] have used methods of KNN algorithm, Naïve Bayes, and Reverse DBSCAN algorithm with experimentation on dataset. For the text recognition, OCR library" [iii] is employed but this OCR doesn't perform well. Mohamad & Selamat (2015) [v] uses the feature selection hybrid approach of TF-IDF (Term Frequency Inverse Document Frequency) and Rough pure mathematics.

*Data Set*

This model has used email data sets from different online websites like Kaggle, sklearn and some data sets are created by own. A spam email data set from Kaggle is used to train our model and then other email data set is used for getting result "spam.csv" data set contains 5573 lines and 2 columns and other data sets contains 574,1001,956 lines of email data set in text format.

## III. METHODOLOGY

### *A. Data preprocessing:*

When the data is considered, always a very large data sets with large no. of rows and columns will be noted. But it is not always the case the data could be in many forms such as Images, Audio and Video files Structured tables etc. Machine doesn't understand images or video, text data as it is, Machine only understand 1s and 0s.

Steps in Data Preprocessing:

*Data cleaning*: In this step the work like filling of "missing values", "smoothing of noisy data", "identifying or removing outliers ", and "resolving of inconsistencies is done."

*Data Integration:* In this step addition of several databases, information files or information set is performed.

*Data transformation:* Aggregation and normalization is performed to scale to a specific value

*Data reduction:* This section obtains a summary of the dataset which is very small in size but so far produces the same analytical result

1. *Stop words*:

"Stop words are the English words that do not add much meaning to a sentence." They can be safely ignored without forgoing the sense of the sentence. For example if it is tried to search a query like" How to make a veg cheese sandwich", the search engine will try to search the web pages that contains the term "how", "to" ,"make", "a" ,"veg", "cheese" ,"sandwich". The search engine tries to find the web pages that contains the term "how" ,"to", "a" than page containing the recipes of veg cheese sandwich because the terms " how" ,"to", "a" are so commonly used in English language ,If these three words are removed or stopped and actually focuses on retrieving pages that contains the keyword " veg", "cheese", "sandwich" – that would give the result of interest.

2. *Tokenization:*

"Tokenization is the process of splitting a stream of manuscript into phrase, symbols, words, or any expressive elements named as tokens." The rundown of token further utilized for contribution for additional handling, for example, content mining and parsing. Tokenization is valuable in both semantics (where it is as content division), and as lexical examination in software engineering and building. It is occasionally hard to define what is intended by the term "word". As tokenization happens at the word level. Frequently a token trusts on modest heuristics, for instance: Tokens are parted by whitespaces characters, like "line break" or "space", or by "punctuation characters". Every single neighboring string of alphabetic characters are a piece of one token; similarly, with numbers.

*3. Bag of words*:

"Bag of Words (BOW) is a method of extracting features from text documents. Further these features can be uses for training machine learning algorithms. Bag of Words creates a vocabulary of all the unique words present in all the document in the Training dataset."

B. *CLASSIC CLASSIFIERS*

Classification is a form of data analysis that extracts the models describing important data classes. A classifier or a model is constructed for prediction of class labels for example:

"A loan application as risky or safe."

Data classification is a two-step

- learning step (construction of classification model.)
- classification step

*1. NAÏVE BAYES:*

Naïve Bayes classifier was used in 1998 for spam recognition. The Naïve Bayes classifier algorithm is an algorithm which is used for supervised learning. The Bayesian classifier works on the dependent events and works on the probability of the event which is going to occur in the future that can be detected from the same event which occurred previously. Naïve Bayes was made on the Bayes theorem which assumes that features are autonomous of each other. Naïve Bayes classifier technique can be used for classifying spam emails as word probability plays main role here. If there is any word which occurs often in spam but not in ham, then that email is spam. Naive Bayes classifier algorithm has become a best technique for email filtering. For this the model is trained using the Naïve Bayes filter very well to work effectively. The Naive Bayes always calculates the probability of each class and the class having the maximum probability is then chosen as an output. Naïve Bayes always provide an accurate result. It is used in many fields like spam filtering.

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

$$P(B) = \sum_{y} P(B|A)P(A)$$

*2. SUPPORT VECTOR MACHINE:*

"The Support Vector Machine (SVM) is a popular Supervised Learning algorithm, the Support Vector model is used for classification problems in Machine Learning techniques. "The Support Vector Machines totally founded on the idea of Decision points. The Main resolution of Support Vector Machine algorithm is to create the line or decision boundary. The Support Vector Machine algorithm gives hyperplane as a output which classifies new samples. In 2- dimensional space "hyperplane is line dividing a plane into 2 parts where each class is present in one side."

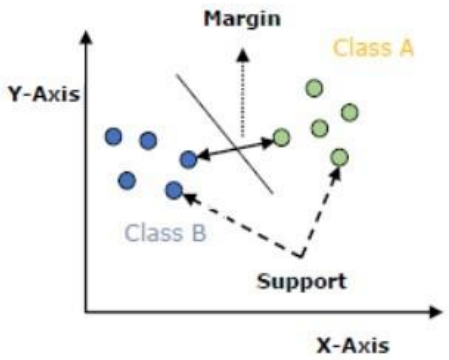


Fig.2 Support Vector Machine

*3. DECISION TREE*:

"Decision tree induction is the learning of decision tree from class labeled training tuples". A decision tree is a flow chart like construction, where

Internal node or non- leaf node= Test on attribute

Branch = shows outcome of the test

Leaf node= holds a class label

Top node is called root node.

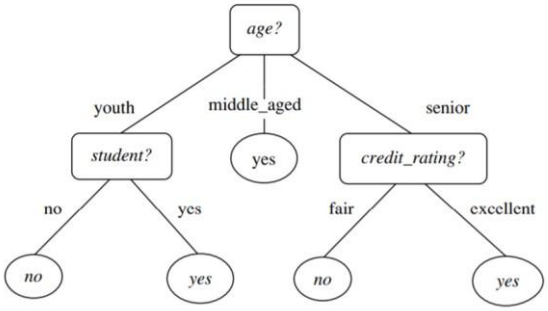



Fig3. Decision Tree Structure

*Decision tree Induction:*

"It handles multidimensional information. the learning and classification phases of decision tree induction are simple and fast. Characteristic choice events are utilized to choose the characteristic that top parcel the tuple into particular classes. At the point when choice tree is manufactured a significant number of the branches may result may reflect commotion and anomalies in the

preparation information. tree pruning endeavors to recognize and evacuate such branches, with the objective of improving classifier precision on an inconspicuous information.

Entropy using the frequency table of one attribute:

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

*4. K- NEAREST NEIGBOUR*

"K-nearest neighbors is a supervised classification algorithm. This algorithm has some data point and data vector that are separated into several classes to predict the classification of new sample point."

K- Nearest neighbor is a LAZY algorithm LAZY algorithm means it tries to only memorize the process it doesn't learn by itself. It doesn't take its own decision by itself.

K- Nearest neighbor algorithm classifies new point based on a similarity measure that can be Euclidian distance.

The Euclidean distance measure Euclidian distance and identifies who are its neighbors.

$$dist((x, y), (a, b)) = \sqrt{(x - a)^2 + (y - b)^2} \quad (5)$$

*C. ENSEMBLE LEARNING METHODS*

"Ensemble methods in machine learning is a method that takes several base model to produce a predictive model in order to decrease. "variance by using bagging bias by using boosting predictions using stacking. Two Types Sequential- here base classifier are created sequentially Parallel- here base classifiers are in parallel.

*1. RANDOM FOREST CLASSIFIER*

Random forest classifier is an ensemble tree classifier consisting of different types of decision trees that are of different shape and sizes.

The random sampling of the training data when building a tree. A random subgroups of input features when splitting at node in a tree. If you have randomness, the randomization will make look the decision tree less corelated so that generalization error

(features of the tree should not look same) of ensemble can be improved.

*2. BAGGING*

"Bagging classifier is an ensemble classifier that fits base classifiers each on random sub sets of the original data sets and then combined their individual calculations by voting or by averaging) to form a final prediction. "Bagging is a mixture of bootstrapping and aggregating.

Bagging= **B**ootstrap **Aggregating**

Bootstrapping helps to lessening the variance of the classifier and it also decline the overfitting by just resampling the data from the training data with same cardinality as in original data set. High variance is not good for the model. Bagging is very effective method for limited data, and by just using samples you are able to get estimate by aggregating the scores.

*3. BOOSTING AND ADABOOST CLASSIFIER*

"Boosting is a ensemble method that is use to create a strong classifier using a number of weak classifier. Boosting is complete by creation a model from a training data sets, then create another model that will precise the faults of the first model." [8] In Boosting Model are added till the training set is predicted properly.

AdaBoost= **Ada**ptive **Boost**ing

AdaBoost is a first fruitful boosting algorithm that was settled for binary classification. The boosting is understood by using AdaBoost.

## IV. ALGORITHMS

**1.1.** Insert the dataset or file for training or testing.

**1.2.** Check the dataset for supported encoding.

**1.2.1.** If one of the supported encodings, then go to step **1.4.**

**1.2.2.** If not one of the supported encoding, then go to step **1.3.**

**1.3.** Change the encoding of the inserted file into one of the supported encodings. Then try again for reading.

**1.4.** Select whether you want to "Train", "Test' or "Compare" the models using the dataset. **1.4.1.** If "Train" is selected, then go to step **1.5.**

**1.4.2.** If "Test" is selected, then go to step **1.6.**

**1.4.3.** If "Compare" is selected, then go to step **1.7.**

**1.5.** "Train" selected:

**1.5.1.** Select which classifier to train using the inserted dataset.

**1.5.2.** Check for duplicates and NAN values.

**1.5.3.** Find the values from Hyperparameter Tuning.

**1.5.4.** Process the text for feature transform.

**1.5.5.** Train the model

**1.5.6.** Save the model and features. Show the results.

**1.5.7.** Select which classifier to test using the inserted dataset.

**1.5.8.** Check for duplicates and NAN values.

**1.5.9.** Load the model and features saved in the training phase of the model.

**1.5.10.** Using the loaded values for testing the dataset.

**1.5.11.** Show the results

**1.6.** "Compare" selected:

**1.6.1.** Compare all the classifiers using the inserted dataset.

**1.6.2.** Show the results of the classifiers.

*A. Implementation*

Visual studio code platform is used to implement the model and, in this module, a dataset from "Kaggle" website is used as a training dataset. The inserted dataset is first checked for duplicates and null values for better performance of the machine. Then, the dataset is split into 2 sub-datasets; say "train dataset" and "test dataset" in the proportion of 70:30. Then the "train" and "test" dataset is then passed as parameters for text-processing. In text-processing, punctuation symbols and words that are in the stop words list are removed and returned as clean words. These clean words are then passed for "Feature Transform". In feature transform, the clean words which are returned from the text-processing are then used for 'fit' and 'transform' to create a vocabulary for the machine.

*B. FlowChart of the model*

The dataset is also passed for "hyperparameter tuning" to find optimal values for the classifier to use according to the dataset.

After acquiring the values from the "hyperparameter tuning", the machine is fitted using those values with a random state. The state of the trained model and features are saved for future use for testing unseen data.

Using classifiers from module sklearn in python, the machines are trained using the values obtained from above.

## V. RESULTS

Our model has been trained using multiple classifiers to check and compare the results for greater accuracy. Each classifier will give its evaluated results to the user. After all the classifiers return its result to the user; then the user can compare it with other results to see whether the data is "spam" or "ham". Each classifier result will be shown in graphs and tables for better understanding. The dataset is obtained from "Kaggle" website for training. The name of the dataset used is "spam.csv". To test the trained machine, a different CSV file is developed with unseen data i.e. data which is not used for the training of the machine; named "emails.csv". After the text edit has been completed, the paper is ready for the template. Duplicate the template file by using the Save As command, and use the naming convention prescribed by your conference for the name of your paper. In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use the scroll down window on the left of the MS Word Formatting toolbar.

a. score 1: using default parameters
b. score 2: using hyperparameter tuning
c. score 3: using stemmer and hyperparameter tuning
d. score 4: using length, stemmer and hyperparameter tuning

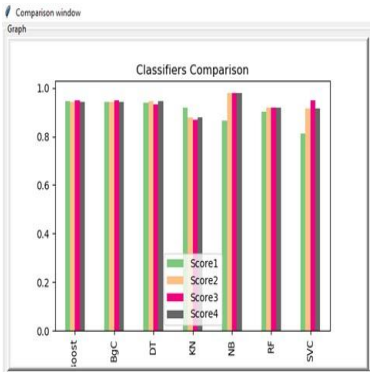
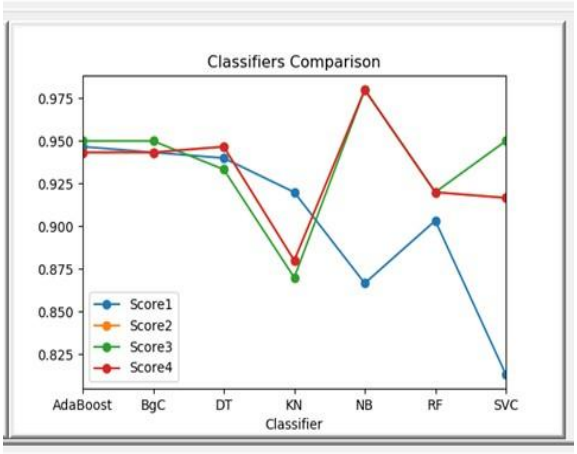



Fig.5 Comparison of all algorithms




## VI . CONCLUSION

With this result, it can be concluded that the Multinomial Naïve Bayes gives the best outcome but has limitation due to class-conditional independence which makes the machine to misclassify some tuples. Ensemble methods on the other hand proven to be useful as they using multiple classifiers for class prediction. Nowadays, lots of emails are sent and received and it is difficult as our project is only able to test emails using a limited amount of corpus. Our project, thus spam detection is proficient of filtering mails giving to the content of the email and not according to the domain names or any other criteria. Therefore, at this it is an only limited body of the email. There is a wide possibility of improvement in our project. The subsequent improvements can be done:

"Filtering of spams can be done on the basis of the trusted and verified domain names."

"The spam email classification is very significant in categorizing e-mails and to distinct e-mails that are spam or non-spam."

"This method can be used by the big body to differentiate decent mails that are only the emails they wish to obtain."

## REFERENCES

[1]. Suryawanshi, Shubhangi & Goswami, Anurag & Patil, Pramod. (2019). Email Spam Detection: An Empirical Comparative Study of Different ML and Ensemble Classifiers. 69-74. 10.1109/IACC48062.2019.8971582.

[2]. Karim, A., Azam, S., Shanmugam, B., Krishnan, K., & Alazab, M. (2019). A Comprehensive Survey for Intelligent Spam Email Detection. IEEE Access, 7, 168261-168295. [08907831]. https://doi.org/10.1109/ACCESS.2019.2954791

[3]. K. Agarwal and T. Kumar, "Email Spam Detection Using Integrated Approach of Naïve Bayes and Particle Swarm Optimization," 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2018, pp. 685-690.

[4]. Harisinghaney, Anirudh, Aman Dixit, Saurabh Gupta, and Anuja Arora. "Text and image-based spam email classification using KNN, Naïve Bayes and Reverse DBSCAN algorithm." In Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conference on, pp.153-155. IEEE, 2014

[5]. Mohamad, Masurah, and Ali Selamat. "An evaluation on t he efficiency of hybrid feature selection in spam email classification." In Computer, Communications, and Control Technology (I4CT), 2015 International Conference on, pp. 227-231. IEEE, 2015