International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 07 | July 2023 www.irjet.net p-ISSN: 2395-0072

Detection of Spam in Emails using Machine Learning

Bhavya V S[1], Yashas R[2], Nithin G M[3], S. Akhila[4]

[1]*Post Graduate Student, Department of ECE, BMS College of engineering, Karnataka, India*
[2]*Post Graduate Student, Department of ECE, BMS College of engineering, Karnataka, India*
[3]*Post Graduate Student, Department of ECE, BMS College of engineering, Karnataka, India*
[4] *Professor, Department of ECE, BMS College of engineering, Karnataka, India*

Abstract - *With fast development of web clients, E-mail spams are increasing alarmingly. People are misusing these spam mails in several ways, to transfer malicious content, unwanted, unsolicited, irrelevant advertisements which can hurt one's framework and spoof on our framework. It could contain malware, such as ransomware and spyware. Creation of a forged or the fake kind of profile and fake email account is far easier for spammers and they create spam mail that is difficult to distinguish from real mail. Thus, it is required to differentiate spam mails and prevent their entry into the inbox. This has been attempted using machine learning techniques. Spam detection through various machine learning algorithms has been attempted and it is found that Multinomial naive Bayes algorithm is more efficient and gives the highest Spam detection with finest accuracy and exactness.*

Key Words: Spam mail, spam detection, machine learning, data set, classifiers

1.INTRODUCTION

The full form of email is electronic mail. Email spam refers to the use of electronic mail to send malicious mail or publicizing mail to gather a recipient's data. These mails are mails that are sent to users who are not the authenticated recipients. These spam mails have caused mishaps on the web by consuming more bandwidth and space. Programmed filtering of mail will be the foremost viable strategy for identifying mail spam, however these days spammers can effortlessly dodge all the applications of spam filtering effectively [1]. Initially, maximum of the spam mails were blocked using Spam filters which have been forwarded from certain email addresses. The most important methods to the spam mail filtering involves investigation of text, domain names boycotts, community and primarily centered techniques. Text assessment of substance sends is a broadly utilized technique to the spams. The approach of machine learning for spam recognition and detection has found to be more efficient compared to the filtering techniques [2]. With emails becoming one of the strategic means of communication, identifying or distinguishing a spam from an authentic mail becomes crucial since these spam mails consume client time and asset producing no valuable yield [3].

Multinomial Naive Bayes is a supreme renowned algorithm connected in the current processes. Though, dismissing sends basically subordinate on dataset investigation can be a challenging matter within the occasion of false positives, frequently the organization and the client do not want any true-blue messages or emails to be misplaced and the reject method has been likely the better strategy sought after, for separation of spams [4]. The procedure is mainly to recognize all the senders out of those from the zone or email ids which are specifically rejected and with latest, the regions which are approaching into the arrangement of spamming space names and this procedure monitors on a work so fine [5]. Another approach is called white list approach, it is the approach of tolerating the sends from the domain names and the addresses straightforwardly whitelisted, putting others in less significant lines. It is conveyed most viable after the sender reacts to a confirmation sent through the junk or the spam mail sifting system [6].

Spam and Ham, concurring from Wikipedia, utilizing electronic mail and informing frameworks to send spontaneous majority mails or messages, particularly frame notice, malicious joins are called spam. Spontaneously implies that the user did not inquire for messages which are coming from the sources [7]. So, on the off chance that the user doesn't know, almost all the sender mail may become spam. People normally don't realize they are fairly marked for those kinds of mailers, when they download any free supervisions and programs while modernizing or updating the program. Ham is the term, which was given by Spam Bayes in the year 2001 and ham is characterized as Emails which are not generally hankered and not considered as spam [8].

The Machine learning methodologies are further effective, the set of training information will be utilized and those tests are set of e-mail which are pre-classified, the machine learning methodologies has section of algorithms which can be utilized for mail sifting and these algorithms incorporate Naive Bayes approach and the support vector machines, Neural Systems, K-nearest neighbour and so on [9][10].

2. LITERATURE SURVEY

In [1] the authors studied the three machine learning methods (SVMs, decision trees, and Naive Bayes) for classifying spam emails. authors developed a supervised classification pipeline to categorise emails as spam or real. The selection of features is one of the key processes in the categorization of spam emails. The top 20 most often used terms in spam and valid emails are selected using the Term Frequency Inverse Document Frequency (TF-IDF) algorithm. The authors studied SVMs, decision trees, and Naive Bayes with specific characteristics to test their effectiveness in identifying and categorising spam email.

In [2] the authors provide a thorough examination of the approaches used in the framework's first component, focusing solely on the domain and header-related data contained in email headers. This work has also looked at a unique feature reduction technique that uses a group of unsupervised feature selection algorithms. In addition, the data source includes a thorough fresh dataset with 100,000 records of spam and junk emails.

In [3] the authors primarily focus on the machine learning-based spam categorization technique. Additionally, authors offer a thorough analysis and assessment of previous research on various machine learning methodologies, email properties, and techniques. Additionally, it outlines potential paths for future study as well as difficulties encountered in the field of spam categorization.

In [4] authors tried to investigate how spam and ham emails may be grouped using unsupervised learning. The basic objective of the project is to create an unsupervised framework that entirely relies on unsupervised techniques using a clustering strategy that uses a number of different algorithms, mostly using the email body and subject header. A unique binary dataset with 22,000 entries of spam and ham emails and 10 features (reduced from eleven to ten after the feature reduction) was used for the clustering. From a multiangular point of view, seven of these ten elements are exclusive to this study and were designed to reflect key analytical email properties.

In[5] authors introduced a technique for spam email detection using machine learning algorithms that are enhanced using bio-inspired techniques. A survey of the literature is conducted to investigate effective techniques used on various datasets to get good outcomes. An intensive study was conducted, along with feature extraction and pre-processing, to build machine learning models utilising Naive Bayes, Support Vector Machine, Random Forest, Decision Tree, and Multi-Layer Perceptron on seven distinct email datasets. Particle Swarm Optimisation and Genetic Algorithm, two bio-inspired algorithms, were used to enhance classifier performance.

3. METHODOLOY

The different classifiers/ algorithms are used to investigate the information of the mails that mainly depicts critical information classes. A classifier is used mainly to build expectation of lesson names like spam or ham. Multinomial Naïve base is a classifier used for spam recognition, it classifies the spam emails and probability of word being in the training set plays fundamental part.

Support vector classifier is used for issues of classification of mails in machine learning, another classifier used is Decision tree which majorly does not require any setting of parameter or domain information for identification and examination of information. This classifier précises on unnoticed data. The random forest classifier consists of diverse sort of choice trees which are of different size and shapes. For datasets with noisy features or unbalanced class distributions, random forests may not be the best option since they can be computationally costly. They are also more difficult to comprehend than individual decision trees. A random forest classifier is often used to generate predictions on fresh, unknown data after being trained on a labeled dataset and having its hyper parameters (such as the number of trees and tree depth) adjusted, K-nearest neighbour, a lazy classifier is another classifier used in the data set model which tries to memorize the method through training since it is not a self learning algorithm, because it does not learn by itself, another classifier is used in the model is Gradient Boosting classifier which is mainly a blend of bootstrap and aggregating, another classifier used in this model is extreme gradient boosting which provides parallel tree boosting and solves ranking problems, another classifier used in the model is adaboost algorithm it mainly creates a new model which will predict the value target variable, another classifier used in the model is information retrieval which is majorly a collection of algorithms which helps in relevancy of presented data.

The first step will be data processing, here the dataset with the number of rows and the columns will be distinguished and here not only the data set containing the case information which also includes the sounds, images, video records. In data processing the steps involved are data cleaning. In data cleaning mainly filling of missing values, smoothing of noise, resolving irregularities will be done. Next process after data cleaning is, data integration in data integration expansion of dataset has been performed, next process is data transformation where normalization is done to scale a particular value, next process is data reduction where it gets mainly rundown of the data set and gets analytical results. After this the data set will be embedded for testing purpose, then the machine will check for the data set for upheld encoding and on the off chance that one of the upheld encoding and select the user needed to train, test or compare the data set. Now on the off chance that will not be

one of the upheld encoding then the machine will alter the encoding of embedded record into backed encodings and at that point only attempts for once more to encode. After the above mentioned process, in the event train is chosen then train will be selected automatically and machine selects which classifier to prepare for utilizing the updated or the inserted data set and it discovers the parameter values and handles the content to highlight and train the model, spares the highlights, so results will appear utilizing the inserted dataset, then checks for not a number (NAN) values and copies, after this highlights the spared in training stage of the model. Next when the event test and compare is chosen firstly it compares the selected and also compares all the classifiers utilizing the inserted data set and finally results of the classifiers are shown.

4. IMPLEMENTATION

The platform which is used to execute the model is visual studio code and in this module, a dataset which is available in the "Kaggle" website has been used as preparing dataset. The dataset inserted to begin with checks for duplicate and invalid values for superior execution of the mechanism. At that point of time, the dataset has been parted in two sub dataset namely, train dataset and test dataset with the range of 60:40. Formerly train and test dataset is conceded as the parameters for text-processing at that point, in text-processing, mainly the punctuation symbols and the words which are within the list of stop words are removed and they are returned back as clean words and these clean words are formerly conceded or passed for feature Transform, In highlight change the clean words which are returned from the process of text-processing are further used for fit and transform to make clear vocabulary for machine and the dataset is additionally passed for tuning of hyper parameter to find out ideal values for the classifier by which dataset is used accordingly. After getting the values from hyper parameter tuning. By using the random values the machine will be best fitted and the unseen data has been stored for further testing. Using the classifiers from the present module which is learned in python, machines are prepared utilizing the values gotten commencing above.

5. RESULTS

The proposed model of ours has been prepared utilizing numerous classifiers to compare and check the results further for noticeable accuracy. Where every classifier will provide its assessed result, after all the classifiers which return its results to the user at that point the user who will compare the obtained result with dataset and with other results to understand whether the information is ham or spam and each classifiers results will be appeared in charts and tables for greater perception and dataset which has been taken as input from Kaggle website for training and title of the dataset utilized is 'emailspamidentify.csv'. To test prepared machines, a diverse common separated value file in short form as csv file has been developed with covered information that is information which is not utilized to train the machine.

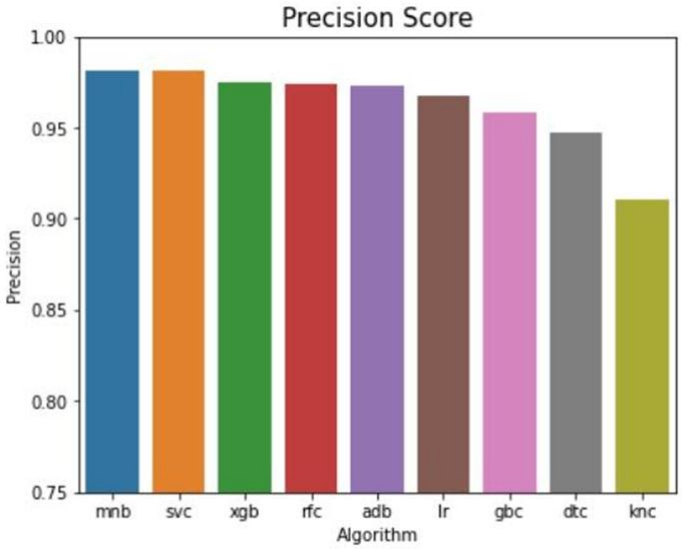


Fig -1: Comparison of all algorithms by using precision score

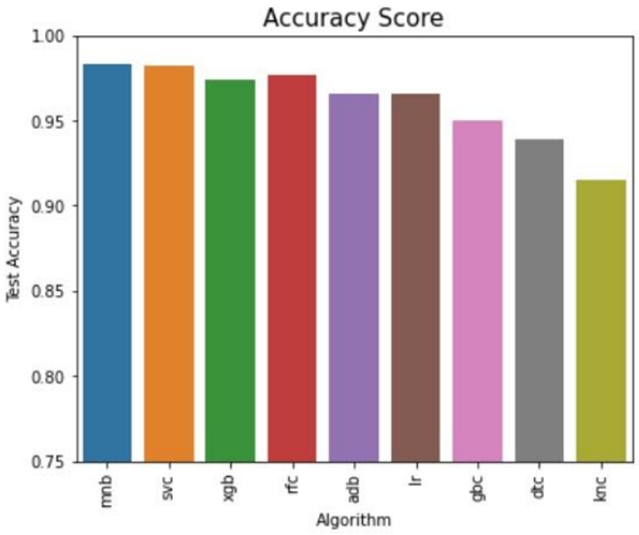
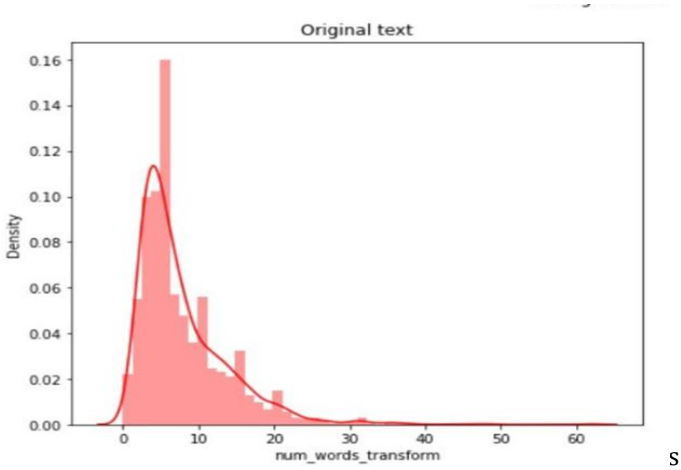


Fig -2: Comparison of all algorithms by using accuracy score



s

Fig -3: Average word length of Original text in email

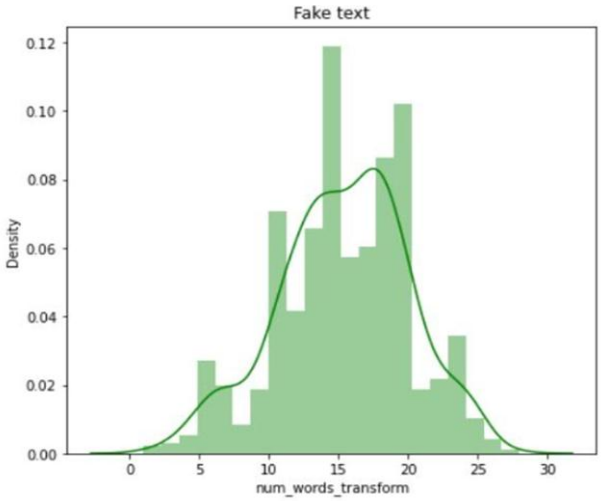


Fig -4: Average word length of fake text in emails

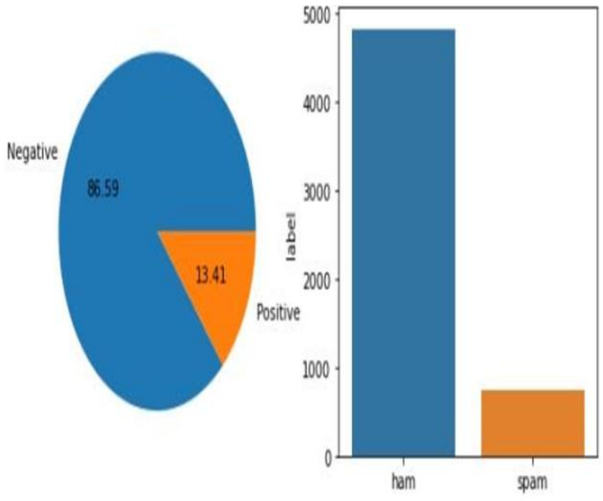


Fig -5: Ratio of ham and spam mails

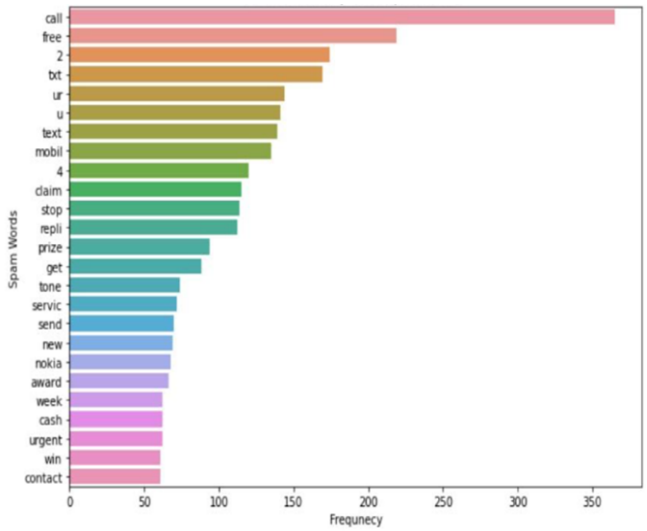
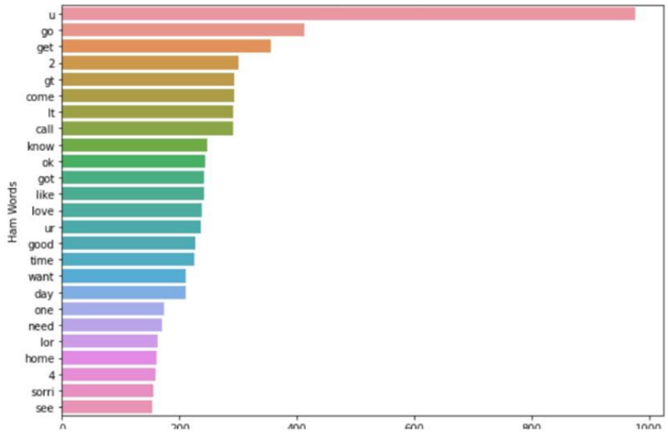


Fig -6: Most Commonly used spam words



n

Fig -7: Most Commonly used ham words

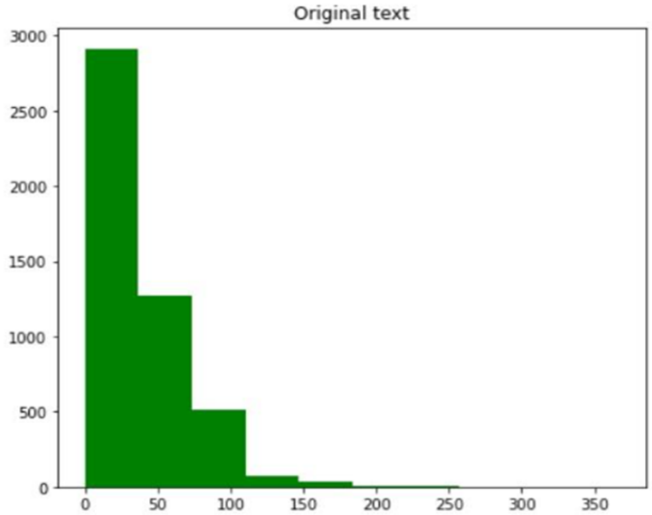


Fig -8: characters in original text

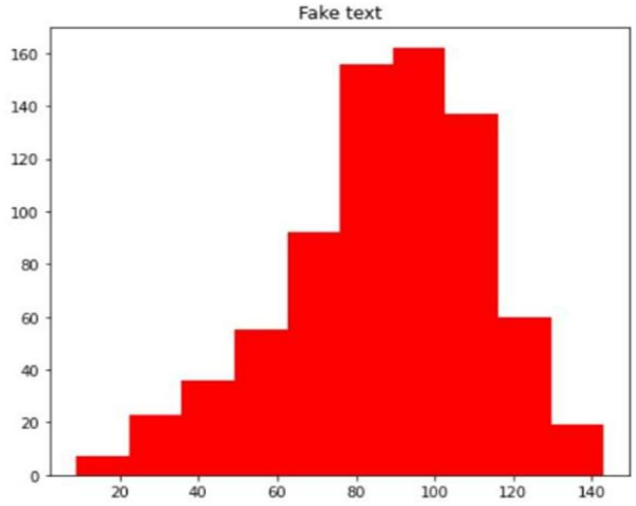


Fig -9: characters in fake text

Table -1: Different classifiers comparison table

CLASSIFIER/ ALGORITHM	PRECISION	TEST ACCURACY	TRAIN ACCURACY
Multinomial Naïve Bayes	0.981	0.983	0.983
Support Vector Classifier	0.981	0.982	0.985
Xtreme Gradient Boosting	0.975	0.974	0.985
Random Forest Algorithm	0.974	0.977	0.985
Adaboost Algorithm	0.973	0.966	0.977
Information retrieval	0.968	0.966	0.968
Gradient Boosting Classifier	0.958	0.950	0.965
Decision Tree Algorithm	0.947	0.939	0.951
K-nearest Neighbour	0.911	0.915	0.933

6. CONCLUSION

With the obtained outcome, it can be decided that Multinomial Naive Bayes provides the better result but it has a certain limit due to class conditional independence which creates a machine to misclassify a few tuples. Gathering strategies have been proven as valuable as the user utilizes numerous classifiers for the prediction of class. These days, loads of mails are received and sent and it is troublesome as our model is able to test mails utilizing a restricted quantity of corpus, In our model or project, hence the spam location is capable of shifting sends and giving it back to the substance mail and not concurring to space names and any further criteria. Therefore, at present it is only a limited body of mail and There could be a wide-ranging credibility of change in our project and the consequent developments can be done in future.

REFERENCES

[1] J, Cui and X. Li, "Content Based Spam Email Classification using Supervised SVM, Decision Trees and Naive Bayes," ICMLCA 2021; 2nd International Conference on Machine Learning and Computer Application, 2021, pp. 1-4.

[2] A. Karim, S. Azam, B. Shanmugam and K. Kannoorpatti, "Efficient Clustering of Emails Into Spam and Ham: The Foundational Study of a Comprehensive Unsupervised Framework," in IEEE Access, vol. 8, pp. 154759-154788, 2020, doi: 10.1109/ACCESS.2020.3017082.

[3] M. RAZA, N. D. Jayasinghe and M. M. A. Muslam, "A Comprehensive Review on Email Spam Classification using Machine Learning Algorithms," 2021 International Conference on Information Networking(ICOIN),2021,pp.327332,doi:10.1109/ICOIN50884.2021.9334020.

[4] A. Karim, S. Azam, B. Shanmugam and K. Kannoorpatti, "An Unsupervised Approach for Content-Based Clustering of Emails Into Spam and Ham Through Multiangular Feature Formulation," in IEEE Access,vol.9,pp.135186135209,2021,doi:10.1109/ACCESS.2021.3116128.

[5] S. Gibson, B. Issac, L. Zhang and S. M. Jacob, "Detecting Spam Email With Machine Learning Optimized With Bio-Inspired Metaheuristic Algorithms," in IEEE Access, vol. 8, pp. 187914-187932, 2020, doi: 10.1109/ACCESS.2020.3030751.

[6] S. Kaddoura, O. Alfandi and N. Dahmani, "A Spam Email Detection Mechanism for English Language Text Emails Using Deep Learning Approach," 2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2020, pp. 193-198, doi: 10.1109/WETICE49692.2020.00045.

[7] S. Nandhini and J. Marseline K.S., "Performance Evaluation of Machine Learning Algorithms for Email Spam Detection," 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE), 2020, pp. 1-4, doi: 10.1109/ic-ETITE47903.2020.312.

[8] R. Al-Haddad, F. Sahwan, A. Aboalmakarem, G. Latif and Y. M. Alufaisan, "Email text analysis for fraud detection through machine learning techniques," 3rd Smart Cities

Symposium (SCS 2020), 2020, pp. 613-616, doi: 10.1049/icp.2021.0909.

[9] N. Govil, K. Agarwal, A. Bansal and A. Varshney, "A Machine Learning based Spam Detection Mechanism," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), 2020, pp. 954-957, doi: 10.1109/ICCMC48092.2020.ICCMC-000177.

[10] M. S. Haghighi and M. Sahebi, "Accelerated supervised learning to detect spam using feature selection and Apache Spark architecture," 2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS), 2020, pp. 1-6, doi: 10.1109/ICSPIS51611.2020.9349587.