Email Spam Detection Using Machine Learning Algorithms

Janani subramaniyan[1]
UG scholar
Departrment of artificial intelligence & data science
Sir Issac newton college of engineering & technology
Nagapattinam, India
Jananisubramaniyan88@gmail.com

Monisha Ramesh[2]
UG scholar
Departrment of artificial intelligence & data science
Sir Issac newton college of engineering & technology
Nagapattinam, India
Moni06012004@gmail.com

Madhana Baskar[3]
UG scholar ,
Departrment of artificial intelligence
& data science
Sir Issac newton college of
engineering & technology,
Nagapattinam, India
Madhanabaskar6@gmail.com

Abstract - **Email spam has grown significantly in recent years along with the rapid expansion of internet users. They are being used for fraud, phishing, and other unethical and criminal activities. sending harmful links through unsolicited email, which can damage our system and try to access your system. The spammers target those people who are unaware of these frauds and target them by easily creating phony profiles and email accounts. In their spam emails, they pose as a real person. In order to identify spam emails that are fraudulent, this project will use machine learning techniques. This article will cover machine learning algorithms and apply all of these algorithms on our data. Despite the existence of various kinds of electronic communication, such social networking, sending and receiving emails has remained the simplest and fastest method. A significant issue in the realm of computing has been the development in online transactions via email, which has globally led to the rising rate of spam emails. For identifying these undesired spams, a variety of machine learning algorithms are described in this note. Despite the notable advancements in the numbers of the literature studied, no machine learning technique has ever achieved 100% accuracy. Only a small number of features and qualities are used by each algorithm for categorization**

Keywords: **Machine learning, Naïve Bayes, support vector machine-nearest neighbor, random forest, neural networks, Email Classification, Ham, Spam Filter, Features, Spam Detection, Email Client, Ambiguous Output, Email Structure**.

I. INTRODUCTION

The easiest and most popular method of communication for both personal and professional purposes is email. Additionally, it is the quickest way to transfer complicated information between users, including not only text but also attachments like pictures, videos, documents, URLs, and more. By doing away with the costs and hassles associated with more traditional ways of communication like letters and faxes, this kind of communication allows us to save a significant amount of time and money. The email system is fundamental to both the business and academic worlds, to the point where it handles all significant conversations and day-to-day tasks. Since its introduction, emailing has propelled economic progress by bringing operations on a global scale to a new level. It is becoming so commonplace in our daily lives. Because 60% of all email traffic is spam, according to the most recent email server survey study, anti-spam filters must be developed. The current spam filters are designed to identify various types of spam emails according to their attributes. Specifically, email spam is filtered using text categorization technology. However, spammers have discovered a new method for getting over the filters that are now in place: they attach textual material based on images to their emails. This technique is known as "image spam," and it's currently the most advanced form of spam mail that uses obfuscation. Machine learning techniques are more effective since they utilize a pre-classified set of emails as training data samples. Numerous algorithms from machine learning techniques can be applied to email screening. "Naïve Bayes, support vector machines, neural networks, K-nearest neighbor, random forests, etc." are some of these algorithm.

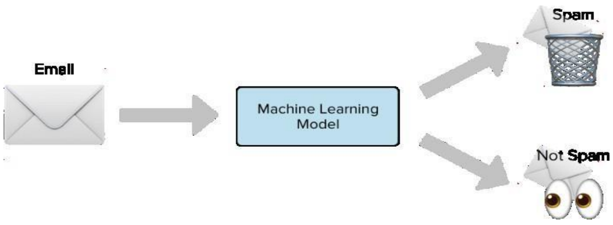


Fig.1. Classification in to Spam and non-spam

II. RELATED STUDIES

In the attention of the international research community, the spike in spam emails is thought to be the cause of the quick rise in email spam filtering. This has prompted researchers to do several comparative studies on the effectiveness of hybridized metrics-based spam image-based email categorization algorithms. Therefore, it's critical to determine which method performs best for a certain statistic in order to enable accurate classification of emails as spam or not. Here, we provide an overview of the relevant and ongoing scientific research efforts that have been published in the literature regarding low-level, OCR-based, and hybrid approaches to email spam filtering based on images.

In order to categorize the textual portion of a picture and classify words in the mail as either spam or non-spam, Chopra et al. [1] used a two-stage technique. The researchers reported in their work titled "The Image and text spam filtering" that spammers have adopted new techniques to incorporate spam email inside the image linked to the package. In the first stage, an OCR tool and a Bayesian algorithm were utilized. The researchers are led to suggest the strategy in an effort to address this issue. Based on the hybridization of KNN and SVM, a method was proposed. The basic idea is to prepare a nearby SVM for the separation task and to categorize the nearest neighbors to a verification challenge. In their study "A process for image spam detection using texture feature," Sadat M. and Rahmati [4] proposed a method in which they identified the spam image by using the image texture function. The co-occurrence gray level matrix (GLCM) was used in this study to apply one of the texture qualities to every picture. The next step is to identify the features that each image has acquired. Both the Bayesian naïve and the neighbours classifier k-nearest are employed. Following the evaluation of the classifiers' 22 acquired properties, the photos from the Dredze and Image

Spam Hunter datasets are assessed. Cross validation techniques split the dataset into training and test sets [4]. The classification's outcome, which took into account four performance metrics—accuracy, precision, recall, and A method that eliminates especially low-level details like picture metadata and histogram features was put forth by Kumaresan T. et al. [5]. In order to detect picture spam, an SVM classifier is applied with the help of a kernel function and the retrieved features. The accuracy of this method is 90%, but the time complexity of the process limits the amount of work that can be done. In this work, classifiers for picture spam were constructed using a variety of image attributes. SVM and PSO together are the classifiers that are employed. PSO ensures that the particles in the search space are moved and iteratively scans candidate solutions to improve the result. Once more, PSO is only easily applicable to datasets that are somewhat small compared to other datasets because of its computational cost. Although it only concentrates on removing unsolicited emails or two-level prioritizing systems, spam filtering is a type of email prioritization. Good results in spam filtering using Naive Bayes classifiers were reported by Sahami et al. Spam filtering does, however, help users with their overload to some extent, and with these adjustments, an email system that produces more accurate and efficient results can be developed. In addition, the goal has been to create a system that generates user-specific output. This guarantees each user of the system will have an optimal user experience.

III. ARCHITECTURE

In the follow-up, this study suggests classifying emails as spam and ham using a machine learning technique, which makes it easier for the algorithm to identify the required elements more precisely rather from having to manually describe them. The primary goal is to categorize user-inputted emails according to a variety of factors that are commonly employed by spammers. Its primary goal is to filter out spam emails and group those that are important. It is a pointless exercise for a system administrator to ban senders from a specified list who are known to send spam because other internet domains are easily accessible and readily available. This work proposes a model that leverages characteristics like the to field, the From field, the Message-ID, the Cc/Bcc field, etc. in order to prioritize machine learning above manual classification. The suggested paradigm has a client-server architecture that is divided into three tiers. The primary function of the first module is data processing. This is where the process of retrieving emails from an email server is initiated. Data formatting is the next step after obtaining the necessary data, which is emails. This enables us to get more precise results. The underlying logic for classifying emails as spam or not is implemented in the second module. Emails with formatting are forwarded to the machine learning library. The next crucial stage is carried out, which is called Explore and Analyze Data (EDA). In this instance, the data is examined with a focus on feature analysis.

This set of features covers the full workflow of the system, from the retrieval procedure to the final concept's classification and spam filtering of the emails. Each stage uses a separate section of the workflow architecture diagram and requires a distinct set of procedures because it is autonomous in terms of functionality but not in terms of the data it requires. The system performs binary classification on emails using a machine learning model that has been trained beforehand. There are some inherent problems with this model. To begin with, a sizable amount of data is required so that the algorithm can "learn" how to classify emails. A wide variety of emails must be included in this dataset in order to maximize the algorithm's accuracy. Second, there is no way that the emails that are filtered out for spam will ever be 100% accurate.

i. Data preparation

Given that the classification algorithm depends on a dataset to carry out its operations, it is critical that emails are retrieved with 100% accuracy from each of their servers, regardless of domain. Protocols like POP3 (Post Office Protocol 3), SMTP (Simple Mail Transfer Protocol), IMAP (Internet Message Access Protocol), and others are used for this retrieval. After the emails are successfully loaded into a PWA or application, the classification problem is the only one left. This is accomplished by reading through the text in the email's header and body in each view. The text files are parsed in order to search for specific keywords and perform a wide spam/non-spam classification. The priority level is then ascertained by the algorithm reading these parameters.

ii. Data analysis

Following the stage of data preparation, the input parameters are taken into account before the algorithm is run. By doing so, the categorization model is "trained" to identify recurring patterns in incoming emails, saving time that would have been needed for additional comparisons. The final algorithm's accuracy is directly correlated with how much training it gets. Surveying the post-training results is necessary for additional analysis. Finding "outliers," or findings that differ from the norm in enough ways to merit a different classification, is the main purpose of this process. The dataset is split into two categories of outcomes by this outlier analysis: proper and ambiguous. The output that is subjected to additional evaluation is the correct one.

iii. Evaluation

A final assessment of the appropriate output is now required before it can be categorized or filtered. The algorithm has already been trained using a set of data that we already have. After getting the user's emails back, we also have an instantaneous outcome. To maximize accuracy, comparisons and contrasts between the dataset findings and the real-time outcomes are required. The same settings are used for another round of training the algorithm. This allows us to further refine the degree and precision of filtering and prioritizing, as well as assess the uncertainty of the output received.

iv. Arrangement

The issue pertaining to the system's practical implementation is the only one left after the emails have been correctly categorized into their appropriate folders. A thorough report must describe and summarize the system's general operation in a way that both specialists and laypeople can understand. The desired pervasiveness of the system for users depends critically on the client's online deployment going well. Eventually, scalability without sacrificing system efficiency becomes essential as the user base grows. Both the method and the PWA ought to be scalable across several devices without sacrificing desired performance.

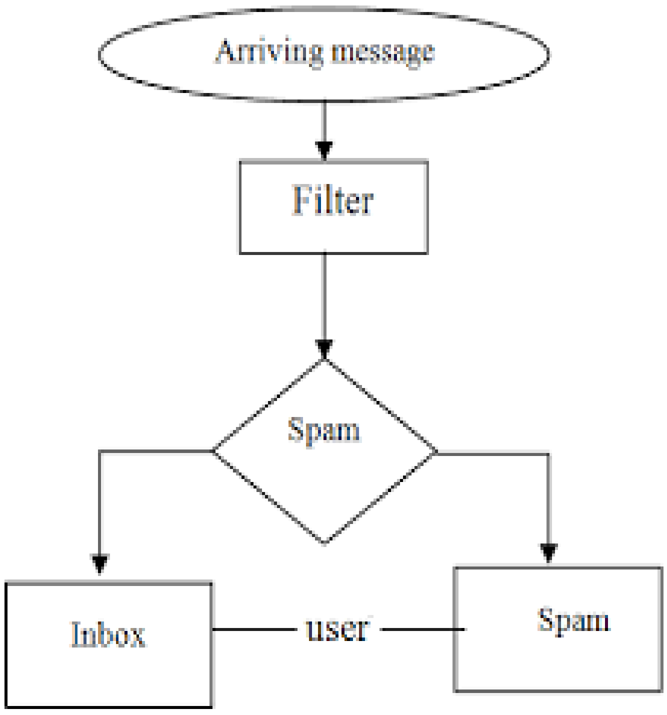


Fig.2. basic structure

IV. METHODOLOGY

The machine only understands 1s and 0s; it cannot comprehend photos, videos, or text data in its current form.

Data Preprocessing Steps:

Data purification:
The tasks of "filling in missing values," "smoothing noisy data ," "identifying or removing outliers," and "resolving inconsistencies" are completed in this step.

Data Integration: This stage involves adding multiple databases, information files, or sets of information. Transformation of data: The purpose of aggregation and normalizing is to scale to a given value.

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.

Data classification has two-step

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

1) NAÏVE BAYES:

In 1998, the Naïve Bayes classifier was employed to identify spam. One method for supervised learning is the Naïve Bayes classifier algorithm. The Bayesian classifier operates on dependent events and calculates the likelihood that an event will occur in the future and may be identified from an earlier occurrence of the same event. The foundation of Naïve Bayes is the Bayes theorem, which posits that features are independent of one another . It can be applied to the classification of spam emails, where word probability is the primary factor. An email is considered spam if any word appears frequently in it but not in ham. The Naive Bayes classifier algorithm is now the most effective method for filtering emails. In order for the model to function well, it is trained using the Naïve Bayes filter. Every time, the Naive Bayes algorithm determines the likelihood of each class; the class with the highest likelihood is then selected as the output. Every time, Naïve Bayes yields an accurate result.

$$\mathrm{P(A|B)} = \frac{P(B|A)P(A)}{P(B)} \qquad (1)$$

$$\mathrm{P(B)} = \sum\nolimits_{y} P(B|A)P(A) \qquad (2)$$

2) SUPPORT VECTOR MACHINE:

"The Support Vector Machine (SVM) is a widely used Supervised Learning algorithm that is employed in machine learning techniques for classification problems. "The concept of decision points serves as the foundation for Support Vector Machines.

The creation of the line, or decision boundary, is the primary resolution of the support vector machine algorithm. The result of the Support Vector Machine method is a hyperplane, which is used to classify new samples.

A "hyperplane is a line dividing a plane into 2 parts where each class is present in one side" in two-dimensional space.

SVM is a powerful supervised algorithm that works best on smaller datasets but on complex ones. Support Vector Machine, abbreviated as SVM can be used for both regression and classification tasks, but generally, they work best.

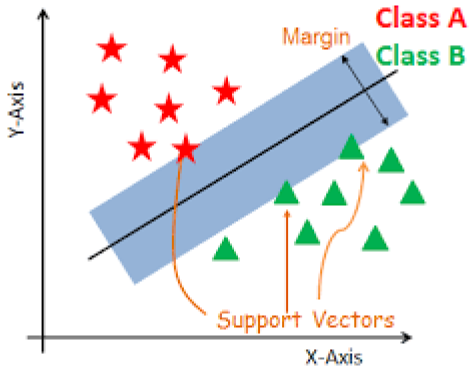


Fig.3. Support Vector Machine (SVM)

3) DECISION TREE:

A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. It has a hierarchical, tree structure, which consists of a root node, branches, internal nodes and leaf nodes. A decision tree is a tree-like structure that represents a series of decisions and their possible consequences. It is used in machine learning for classification and regression tasks. An example of a decision tree is a flowchart that helps a person decide what to wear based on the weather conditions. The main components of a decision tree include a root node, decision nodes, chance nodes, alternative branches, and an endpoint node. Optional features include rejected alternatives.

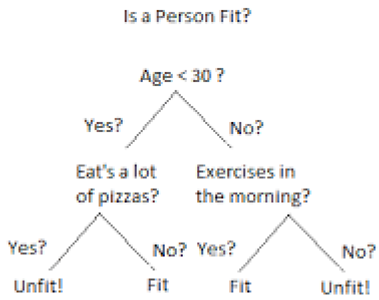


Fig.4. Decision Tree Structure

4) K- NEAREST NEIGBOUR:

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data 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.

$$\mathrm{dist((x,\ y),\ (a,\ b))} = \sqrt{(x - a)^2 + (y - b)^2} \qquad (3)$$

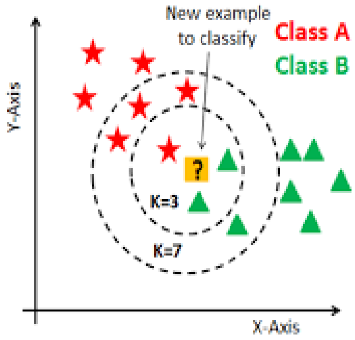


Fig.5. K- Nearest neighbor

Ensemble learning techniques:

1. *RANDOM FOREST CLASSIFIER:*

The Random forest classifier creates a set of decision trees from a randomly selected subset of the training set. It is basically a set of decision trees (DT) from a randomly selected subset of the training set and then It collects the votes from different decision trees to decide the final prediction.

2. *BAGGING:*

Bagging, also known as bootstrap aggregation, is the ensemble learning method that is commonly used to reduce variance within a noisy dataset. In bagging, a random sample of data in a training set is selected with replacement—meaning that the individual data points can be chosen more than once.

Bagging is used with decision trees, where it significantly raises the stability of models in improving accuracy and reducing variance, which eliminates the challenge of overfitting. Bagging in ensemble machine learning takes several weak models, aggregating the predictions to select the best prediction .In fact, an example of the bagging technique is the random forest algorithm. The random forest is an ensemble of multiple decision trees. Decision trees tend to be prone to overfitting. Because of this, a single decision tree can't be relied on for making predictions.

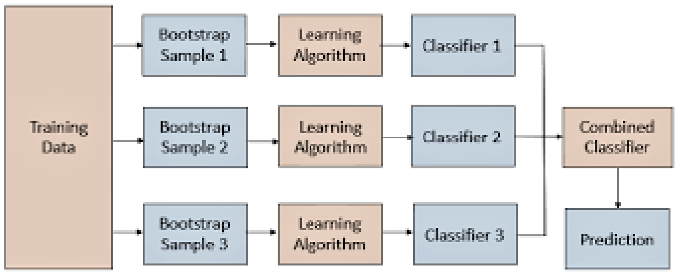


Fig.6. Bagging

3. *BOOSTING AND ADABOOST CLASSIFIER:*

Boosting is an ensemble modeling technique that attempts to build a strong classifier from the number of weak classifiers. It is done by building a model by using weak models in series. Firstly, a model is built from the training data. Boosting is an algorithm that helps in reducing variance and bias in a machine learning ensemble. The algorithm helps in the conversion of weak learners into strong learners by combining N number of learners. Boosting also can improve model predictions for learning algorithms.

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.

The basic concept behind Adaboost is to set the weights of classifiers and training the data sample in each iteration such that it ensures the accurate predictions of unusual observations. Any machine learning algorithm can be used as base classifier if it accepts weights on the training set.

V. ALGORITHM

Stage 1. Training

```
Parse each email into its constituent tokens
 Generate a probability for each token
 W $S[W] = Cspam(W) / (Cham(W) + Cspam(W))$
 store spamminess values to a database
```

Stage 2. Filtering

```
 For each message
M while (M not end) do
scan message for the next token Ti
query the database for spamminess S(Ti)
 calculate accumulated message probabilities S[M] and H[M]
 Calculate the overall message filtering indication by:
$I[M] = f(S[M] , H[M])$
f is a filter dependent function
such as $I[M] = 1+S[M]-H[M]/ 2$
 if $I[M] >$ threshold
msg is marked as spam
else
 msg is marked as non-spam
```

VI. RESULT

The training dataset, spam and legitimate message corpus is generated from the mails that we received from our institute mail server for a period of six months. The mails are analyzed and 23 rules are identified that extremely ease the process of classifying the spam message. The corpus consists of 750 spam messages and 750 legitimate messages. From the corpus, the feature vectors are extracted by analyzing message header, keyword checking, white list/blacklist etc.

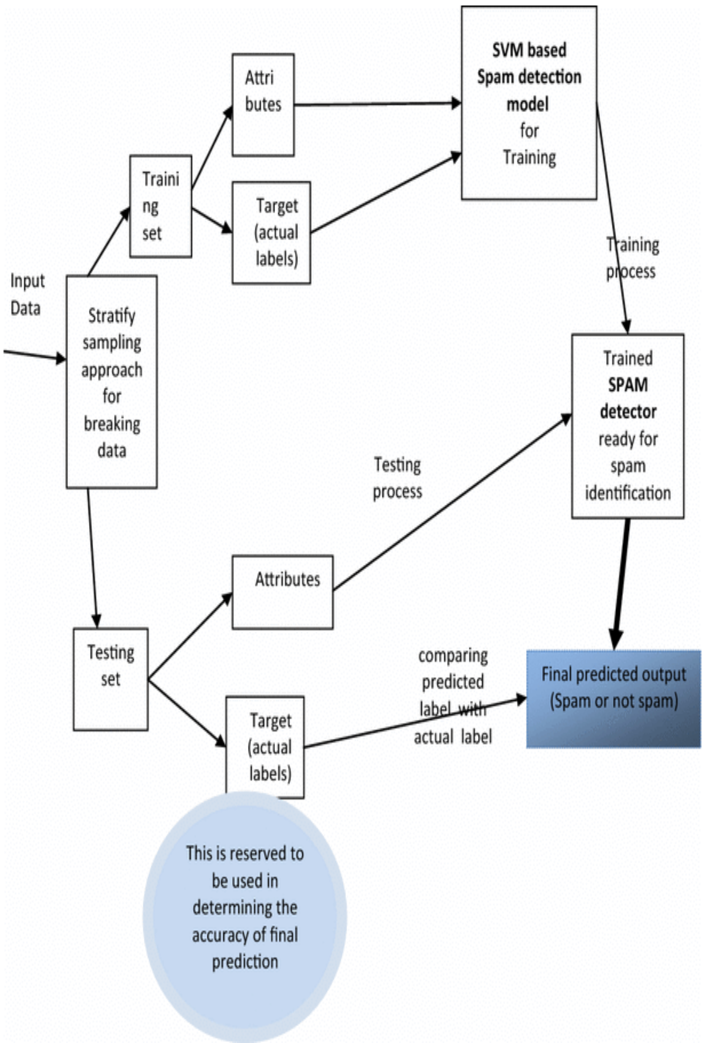
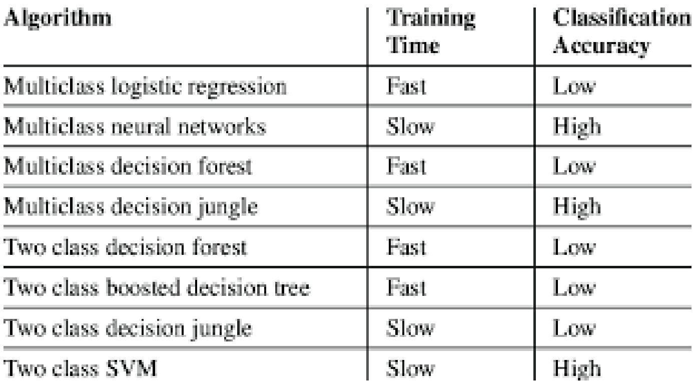

Input Data
Stratify sampling approach for breaking data
Traini ng set
Attri butes
Target (actual labels)
SVM based Spam detection model for Training
Training process
Trained SPAM detector ready for spam identification
Testing process
Attributes
Testing set
Target (actual labels)
comparing predicted label with actual label
Final predicted output (Spam or not spam)
This is reserved to be used in determining the accuracy of final prediction

Fig.7. Flow chart for spam email classification

VI. ALGORITHMS COMPARISON

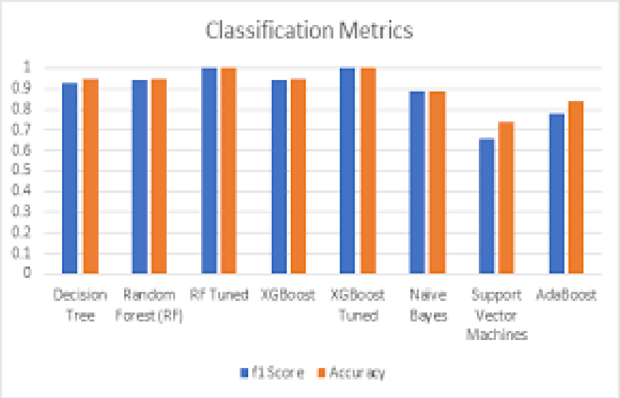
	Classifiers	Score 1	Score 2	Score 3	Score 4
1	Support Vector Classifier	0.81	0.92	0.95	0.92
2	K-Nearest Neighbour	0.92	0.88	0.87	0.88
3	Naïve Bayes	0.87	0.98	0.98	0.98
4	Decision Tree	0.94	0.95	0.93	0.95
5	Random Forest	0.90	0.92	0.92	0.92
6	AdaBoost Classifier	0.95	0.94	0.95	0.94
7	Bagging Classifier	0.94	0.94	0.95	0.94



Algorithm	Training Time	Classification Accuracy
Multiclass logistic regression	Fast	Low
Multiclass neural networks	Slow	High
Multiclass decision forest	Fast	Low
Multiclass decision jungle	Slow	High
Two class decision forest	Fast	Low
Two class boosted decision tree	Fast	Low
Two class decision jungle	Slow	Low
Two class SVM	Slow	High

Fig.8.Algorithms time &accuracy

VII. CLASSIFICATION METRICES



VIII. CONCLUSION

Based on this finding, it can be said that while the Multinomial Naïve Bayes algorithm produces the best results, it is limited by class-conditional independence, which causes the system to incorrectly categorize certain tuples. On the other hand, ensemble approaches have shown to be beneficial since they predict classes using a number of classifiers. These days, a lot of emails are sent and received, making it challenging because our project can only examine a small corpus of emails. Because of this, our project's spam detection is capable of filtering emails based solely on their content, not on domain names or other factors. As such, this is just a portion of the email's body. Our project has a great deal of room for improvement. The following enhancements are possible:

"Spam can be filtered using trusted and verified domain names as a basis."

"Classifying spam emails is crucial for both classifying emails and differentiating between spam and non-spam emails."

"The big body can use this method to distinguish between good emails and emails that are only what they want to receive."

A secure email client that automatically rejects spam emails is vital for individual users. As the amount of the dataset increases, a self-learning system that is tailored to each user and based on their dataset will only guarantee increased accuracy. As a result, the system gets closer to finding the ideal solution over time

IX. FUTURE WORKS

For the purpose of identifying these unsolicited spams, numerous machine learning approaches are available. Despite notable advancements in the amount and quality of reviewed literature, no machine learning technique has attained 100% accuracy. Only a few features and qualities are used by each algorithm for classification. Because of this, choosing the optimal algorithm is a crucial effort that requires balancing its advantages over disadvantages. Therefore, further study is needed to enhance the hybrid system's performance on the artificial immune system and to concentrate on the availability of well-labeled datasets to guarantee efficient spam filtering. Additionally, it has been observed that there is a surge in the use of the internet and that this use and application is correlated with the proliferation of spam images. Spam emails are more than just a harmless time waster. It is a tool for harmful actions like website forgeries, spear phishing, whaling, and clone phishing, among many others. Identifying emails as spam or ham is therefore crucial from the user's security point of view. Thus, the suggested system uses a previously classified dataset to train the algorithm and categorize emails. It then expands on that functionality to classify incoming emails and present them in an orderly fashion. By decreasing the clutter and distractions brought about by spam, this not only increases productivity but, more crucially, shields the user from malicious attacks. Considering how many dangers there are, it is critical for security to use this straightforward yet powerful strategy.

X. REFERENCES

[1] Ravikumar K, Gandhimathi P. A (2014) Review on Different Spam Detection Approaches.

[2] Renuka, D.K.; Visalakshi, P.; Sankar, T.J.I.J.C.A. Improving E - mail spam classification using ant colony optimization algorithm. Int. J. Comput. Appl. 2015, 2, 22–26.

[3] Kumaresan, T., Sanjushree, S., Suhasini, K. and Palanisamy, C., (2015). Image spam filtering using support vector machine and particleswarm optimization. Int. J. Comput. Appl, 1, pp.17 -21.

[4] Wang, Jianyi, and Kazuki Katagishi (2014) "Image Content -Based" Email Spam Image" Filtering." Journal of Advances in Computer Networks 2, no. 2: 110-114.

[5] Das, Meghali, and Vijay Prasad (2014). "Analysis of an Image Spam in Email Based on Content Analysis." International Journal on Natural Language Computing (IJNLC) 3, no. 3, pp. 129 -140.

[6] Liu, Tzong-Jye, Cheng-Nan Wu, Chia-Lin Lee, and Ching-Wen Chen (2014). "A self-adaptable image spam filtering system." Journal of the Chinese Institute of Engineers 37, no. 4, pp. 517 -528.

[7] Foqaha, Mohammed Awad1 and Monir.(2016) "EMAIL SPAM CLASSIFICATION USING HYBRID APPROACH OF RBF NEURAL NETWORK AND PARTICLE SWARM OPTIMIZATION.".

[8] J. M. Carmona-cejudo, G. Castillo, M. Baena-garcía, and R. Morales-bueno, "Knowledge-Based Systems A comparative study on feature selection and adaptive strategies for email foldering using the ABC-DynF framework," vol. 46, pp. 81–94, 2013.

[9] 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.

[10] 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.

[11] 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

[12] D.M. Fonseca, O.H. Fazzion, E. Cunha, I. Las-Casas, P.D. Guedes, W. Meira, M. Chaves, Measuring characterizing, and avoiding spam traffic costs, IEEE Int. Comp. 99 (2016).

[13] A. Bhowmick, S.M. Hazarika, Machine Learning for E-Mail Spam Filtering: Review, Techniques and Trends, arXiv:1606.01042v1 [cs.LG] 3 Jun 2016, 2016, pp. 1–27.

[14] Available at, Mail Server Solution, 2017, http://telco-soft.in/mailserver.php.

[15] S. Dipika, D. Kanchan, Spam e-mails filtering techniques, Int. J. Tech. Res. Appl. 4 (6) (2016) 7–11.

[16] Z.S. Torabi, M.H. Nadimi-Shahraki, A. Nabiollahi, Efficient support vector machines for spam detection: a survey. (IJCSIS), Int. J. Comput. Sci. Inf. Secur. 13 (1) (2015) 11–28.

[17] Al-Duwairi, Basheer, Ismail Khater, and Omar Al-Jarrah (2012).