JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

SPAM EMAIL DETECTION USING MACHINE LEARNING APPROACHES

Dr. K. Anandan
Assistant professor, Department of Computer Applications ,
Nehru College of Management ,Coimbatore

M. Karthikeyan
II MCA student, Department of Computer Applications,
Nehru College of Management, Coimbatore, Tamilnadu, India

Abstract-- As Email became a major platform for professionals and all other people for communicating and sharing information to each other, there will be higher chances of receiving spam mails such as mails that are not relevant and unrelated to the recipients and also illegal mails sent by attackers. Which causes a trouble to the recipients. So in order to identify those spam mails, we can create a machine learning model to identify spam mails. This project aims to create a basic model using several algorithms and find out which algorithm suits best to create a powerful and effective model to identify spam mails. This paper discuss the machine learning algorithms used to make the project and which suits best to identify the spam mails based on its precision and accuracy.

Keywords: *spam,machine learning,algorithms,accuracy.*

I. INTRODUCTION

Email is a means of communication via people over the internet. We can communicate with people across the world from their devices. Emails are mostly used for professional communication and are almost used by all people. Since email is majorly used by people to share information via the internet, there is a potential risk of getting spam mails. As the use of email increased, the probability of getting spam mails also increased. Spam mails are inevitable; every recipient gets spam mails in their everyday life. They are totally worthless and just jam the storage. The spam mail sender will collect the email addresses from different sources over the internet, such as filled forms, etc. Spam mails can be sent for various reasons, such as sending viruses, phishing attacks, and much more. Spam mails are mails that are unwanted or unrelated to the recipients. Those are typically about advertising, sending multiple mails to advertise and promote.

Spam and ham: Spam mails are unwanted and unsolicited messages that are aimed at serving commercial or fraudulent purposes without the recipient's consent. On the other hand, ham mails are genuine messages that are sent by the authorized or trusted sources. The sender of those mails can be identified easily, and they are intended to send those messages to the specific recipient. The difference between spam and ham is that ham is intended to communicate with the recipient, and spam is intended to promote or scam the recipient.

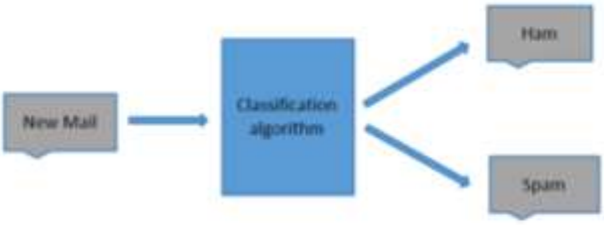


Fig.1 Spam and ham classification

Instead of manually finding the spam mail, we use automatic spam filtering methods, but nowadays those are easily bypassed. So we can create a machine learning model that identifies the spam mails that are effective these days, considering the growth of the technologies. In this project we use multiple machine learning algorithms, which are all supervised algorithms. By creating a model, we can find which model will be accurate to classify the spam and ham mails effectively. The models that are used are, namely, logistic regression, decision trees, support vector machines, and naive bayes.

II. LITERATURE REVIEW

Since the spam mails are increasing researchers started to find out the effective way to filter the spam mails and here are some related works regarding the spam email detection.

Bhuiyan H et al.[2] proposed approcah on A survey of existing e-mail spam filtering method considering machine learning techniques which suggests that the majority of email spam filtering process performed through Machine learning technique using Naive bayes and SVM algorithm are effective and said still there is something lacking and researchers trying to find different process.
Zeeshan Bin Siddique et al.[3] proposed paper on Machine Learning-Based Detection of Spam Emails by where they used machine learning and deep learning and the findings suggests that LSTM is efficient in spam detection with the accuracy of 98% with low model loss rate of 5%.
Mahmoud jazzar et al.[4] paper on Evaluation of Machine Learning Techniques for Email Spam Classification finds that the SVM has the high accuracy despite of its training time its better than the other techniques and also has a lesser False positive rate which makes the SVM take the lead.
The Paper on Machine learning based spam E-mail detection by priti sharma and uma bhardwaj[5], suggests that Naïve Bayes algorithm is simple and also easy to understand and implement and it shows good results even with small amount of training data. But the algorithm works with an assumption of dataset with independent class features. Eventhough naive bayes is good J48 algorithm works better in email spam detection and the hybrid bagged approach got less accuracy.
The Paper on Hybrid spam detection using machine learning by diksha et al.[6]
have shown a result of hybrid method where the Naive bayes and SVM combined to create a model and also separately. By the finding it stated that the hybrid model has more accuracy of 97.57% than the separate models.

III. MACHINE LEARNING APPROACHES

To develop the models we have used 4 supervised ML approaches. The approaches are specified below:

Logistic Regression:

Logistic Regression is majorly used for binary classification which is either 0 or 1 or true or false. It works by analyzing the input features by calculating the belonging to a specific class. It uses a sigmoid function to calculate probabilities. It determines the linear boundary separating the two classes and the sigmoid function maps any real valued input to a value between 0 and 1 representing the probability of belonging to the positive class. When we use a straight line to find the probability the chances of getting free spaces can occur. So instead we use sigmoid curve which can be more precise in finding the right one. So logistic regression technique is more reliable to find out the binary classification.

Support Vector Machine:

This approach work by finding the optimal decision boundary known as the hyperplane that maximizes the margin between the different classes. The space between the hyperplane and the closest data points is said to be margin.
SVM prioritize maximizing the margin for robust and reliable classification. But svm is different from other classification models because the use of kernel function to transform the input data into a higher dimensional space so that the linear separation becomes possible as svm are robust to outliers as they focus on the support vector near the decision boundary rather than every data point and they can handle high dimensional data effectively making them suitable for complex data set with multiple features.

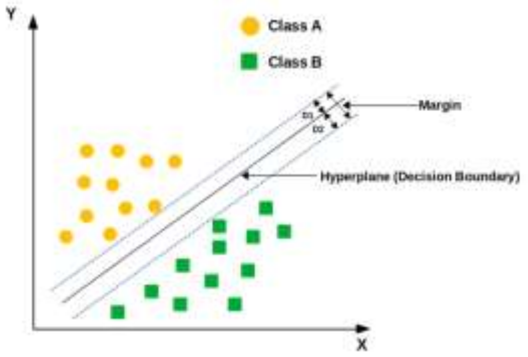


Fig.2 SVM

Decision Tree

This is used for creating classification model. It resembles a tree like structure which can used to predict a class of the variable by simple decision rules. For predicting we start from root of the tree and compare the root attribute by record's attribute.It takes in data and asks questions based on features or characteristics to divide the data into smaller groups. The tree keeps branching based on those answers, leading to a final decision or prediction. The goal is to make the best decision at each step, so that the end result is as accurate as possible. For classifying new emails, the tree is traversed from the root node down to a leaf node, based on the features of the email. The leaf node's label will be the predicted class (spam or ham).

Naive Bayes

The Naive Bayes algorithm helps us predict the probability of an email being spam based on the words that appear in it. It is a fast and simple algorithm that works well for spam detection. While the assumption of feature independence can be unrealistic in some cases, Naive Bayes remains a powerful tool for text classification tasks, particularly when combined with effective feature extraction techniques like Bag of Words or TF-IDF. The algorithm uses the labeled training data to learn the probabilities of each word occurring in spam and ham emails.For a new email, the algorithm calculates the probability of it being spam or ham based on the words in the email.

IV. METHODOLOGY

The process carried out in this research involved three steps: Dataset Preparation, Pre-processing and building various machine learning models which will be used to classify the ham and spam mails.

A. Dataset Preparation

The dataset used in this is obtained from the online site called Kaggle. The mail dataset is the one used in this which contains two categories such as Message and Category. The message contains the actual message and the category contains the category of the message such as Ham or Spam. The dataset contains 5000 plus messages which is used to train and test respectively. Over 80% of the data has been splitted for train the model and the reamining 20% were used to test the trained model to measure the performance.

B. Data pre-processing

The main goal of data pre processing is to get the quality of the data such as handling missing values, incomplete values etc… If the data has a missing values or incomplete values that would result in the model which can possibly affect its accuracy and cause such problems in getting the accurate results as we expected. To solve this we can convert the null values into null string to get on with our process.

Also duplicate data will be removed in the process of data cleaning because it can cause much time and the slow the process. So to gain the possible format to train the model the data preprocessing should be done which involves cleaning the data,transforming the data into accurate format.

C. Test Train split

The dataset obtained will have large number of data which will be divided as two categories such as train and test. A part of the dataset will be taken to train the model and the other will be used to test our trained model. Here the 80% of the data is used to train the data and the remaining is used to test our trained model.

D. Building a Model

Google Colab is used to build the Model. It is a free, cloud-based platform provided by Google that allows to write and execute Python code in a Jupyter notebook environment.It can easily save and load files from Google Drive, which is integrated into Colab. Most of the Python libraries such as TensorFlow, PyTorch, Keras, Pandas, Matplotlib, NumPy, etc., are already pre-installed, which will be a main advantage to use google colab.It provides a straightforward analysis and visualisation of the confusion matrix, true positive, accuracy, recall, false negative, etc.

E. Performance metrics

i. Accuracy

Accuracy measures how well the model is performing with correct predictions.

Accuracy = (True Positives + True Negatives) / (Total Number of Predictions) (1)

ii. Precision

It tells us that how many positive predictions are actually correct that are made by the model. It is calculated by dividing the number of true positives by number of true positives plus number of false positives.

Precision = Number of true positives/Number of true positives + False positives. (2)

iii. Recall

It is the measure of how many true positives that our model is correctly identifying.

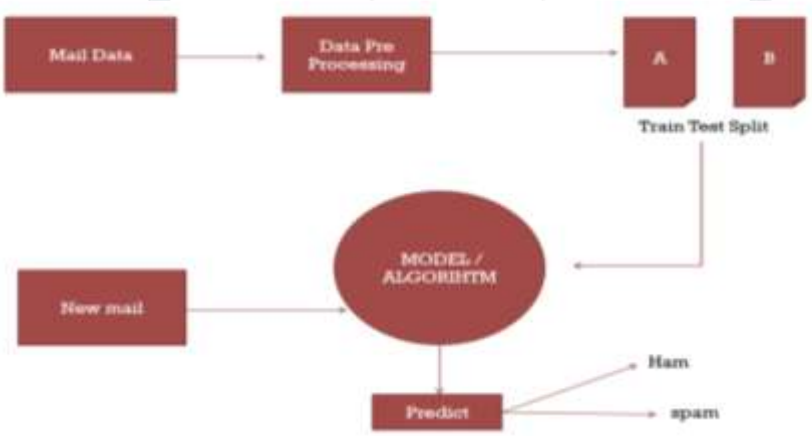
Recall= True positives/True positives + False negative. (3)

iv. F1 score

It is the harmonic mean of recall and precision.

F1 = 2 * (precision * recall) / (precision + recall). (4)

F. Work Flow diagram



V. IMPLEMENTATION

Google Colab platform is used to implement the model and a dataset from “Kaggle” website is used as a training dataset. The dataset is inserted and 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 Then the train and test dataset is then passed as parameters for text-processing. The feature extraction is done to convert the data into numerical form with the help of TF-IDF vectorizer, after the conversion the data will be given to the model. The dataset is also passed for “hyperparameter tuning” to find optimal values for the classifier to use according to the dataset.Then the training of the model will be done with the help of the data that have been converted into numerical form.The fit function is used to fit our model to the dataset which will be helpful to train our model.Then we can evaluate the model whether it predicts accurate or not using the predict function and also can check the accuracy using the sklearn.metrics which will be used to compare our models and find out the accurate one to predict the spam mails.

VI. RESULTS

The model has been trained using different classifiers to check and compare the results for greater accuracy. Each classifier result is evaluated and compared to find the best classifier. The results of each model on accuracy,precision,recall and f1 score are given below in the table.

Table 1. Performance measure of Models

si.no	classifiers	Accuracy	Precision	Recall	F1 Score
1	Logistic regression	0.97	0.98	0.88	0.92
2	Decision Tree	0.96	0.95	0.88	0.91
3	SVM	0.98	0.99	0.93	0.95
4	Naive Bayes	0.97	0.98	0.90	0.94

The comparison of models with the help of graph using matplotlib will be better for understanding the difference. So the graph is displayesd below. Where accuracy is blue in color, precision in orange,recall in green and f1 score in red.

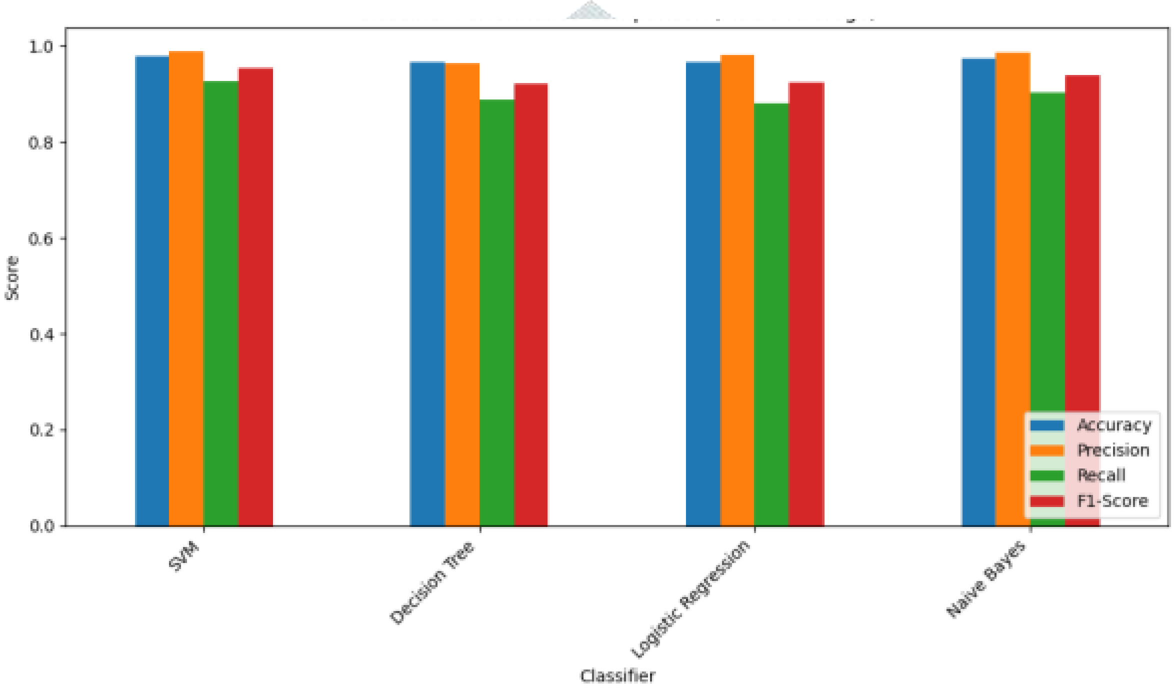


Fig.1 Comparison of Models

VII.CONCLUSION

By this comparative study evaluated four machine learning models—SVM, Decision Tree, Logistic Regression, and Naive Bayes for spam detection. The results indicate that SVM (Support Vector Machine) achieved the highest accuracy, scoring approximately 98%, which shows its potential for accurately classifying emails as spam or ham. While other models like Logistic Regression and Decision Tree demonstrated good performance, SVM emerged as the most effective for this task. This finding underscores the importance of model selection and adaptability to specific dataset characteristics. Future work can explore strategies for fine-tuning SVM parameters or investigating ensemble methods to further enhance classification accuracy. This study underscores the importance of model evaluation and selection for critical tasks like spam detection. The exceptional performance of the SVM, coupled with the insights gained through this research, setting the way for developing more advanced and resilient spam filtering solutions, enhancing cybersecurity and user experience.

Overall,Our findings offer valuable insights into the effectiveness of machine learning for spam detection and contribute towards building robust solutions for email security.

VIII. REFERENCES

[1] Spam email. spam email - an overview | ScienceDirect Topics. (n.d.).

[2] Bhuiyan, H., Ashiquzzaman, A., Juthi, T. I., Biswas, S., & Ara, J. (2018). A survey of existing e-mail spam filtering methods considering machine learning techniques. Global Journal of Computer Science and Technology, 18(2), 20-29.

[3] Siddique, Z. B., Khan, M. A., Din, I. U., Almogren, A., Mohiuddin, I., & Nazir, S. (2021). Machine Learning-Based Detection of Spam Emails. Scientific Programming, 2021(1), 6508784.

[4] Jazzar, M., Yousef, R. F., & Eleyan, D. (2021). Evaluation of machine learning techniques for email spam classification. International Journal Of Education And Management Engineering, 11(4), 35-42.

[5] Sharma, P., & Bhardwaj, U. (2018). Machine Learning based Spam E-Mail Detection. International Journal of Intelligent Engineering & Systems, 11(3).

[6] Jawale, D. S., Mahajan, A. G., Shinkar, K. R., & Katdare, V. V. (2018). Hybrid spam detection using machine learning. International Journal of Advance Research, Ideas and Innovations in Technology, 4(2), 2828-2832.

[7] Anshul. (2025, February 4). Support Vector Machine (SVM) algorithm. AnalyticsVidhya.

[8] Naïve Bayes Algorithm: Everything you need to know. KDnuggets. (n.d.).https://www.kdnuggets.com/2020/06/naive-bayes-algorithm-everything.html

[9] Understanding Micro, Macro, and weighted averages for Scikit-learn metrics in multi-class classification with example. Amir Masoud Sefidian - Sefidian Academy. (2024, November 24).

[10] 8, A. (2025, January 7). Everything you need to know about logistic regression - spiceworks. Spiceworks Inc.

[11] Navlani, A. (2024, June 27). Python decision tree classification tutorial: Scikit-Learn Decisiontreeclassifier. DataCamp.