International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 09 | Sept - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Spam or Junk E-Mail Detection with Help of Machine Learning

Vaibhav Gupta
Department of Computer Science and Engineering
Greater Noida Institute of Technology
Greater Noida . UP
Vaibhavgupta16202@gmail.com

Abstract— In this era of online, we rely exclusively on an email service for short statistics trade however spamming has affected cyber international with inappropriate message that consists of cyber assaults and personal information misuse. The cost of phishing and spam emails, which costs businesses and individuals millions of dollars each year, has been rising exponentially. Some techniques and models have been introduced and developed for the automatic detection of spam emails, but they have not demonstrated 100% predicative accuracy. The price of junk mails is hovering ordinary. This text especially aims at developing a device learning model for spam detection. Machine studying allows in separating unsolicited mail from excellent emails at a totally high fee. Spam also wastes storage, bandwidth and different sources too. We are able to use one of a kind device studying algorithms including Naïve Bayes, support Vector Machines (SVM) or decision timber to identify spam or non-unsolicited mail (uncooked) emails that may obtain accuracy because these algorithms usually obtain better accuracies. The present paper will speak those strategies, follow them to our statistics and consider the algorithm that offers accuracy and precision.

Keywords— System learning Algorithms, Naïve Bayes (NB) , Support vector machines (SVM), Datasets, Spam Mail , Machine learning Algorithm

I. INTRODUCTION

In this big global, time complexity plays an important position in every factor. Today, we use on the spot electronic Mail, often known as email, to change messages faster. 4 million human beings use this device every day. Therefore it should be pretty safe to use. It has become an integral service for lots of humans at the net. As it is easy and it is straightforward to ship messages to a huge wide variety of human beings. E-mail has come to be the maximum popular at the web. There is a hazard to this system referred to as spam. However, a common challenge with this feature is a relentless stream of spam or junk emails. Spam emails, which are obviously unsolicited and often contain malicious content or unwanted advertisements, not only clutter inboxes but also pose a serious security risk Studies show that everyone's email in spam emails -A large proportion of traffic is present, emphasizing urgency with efficient development methods of detection.

Traditional spam filtering methods that rely on rule-based or heuristic methods often struggle to keep up with changing spam patterns. In contrast, machine learning (ML) offers a promising way to combat spam, offering to learn from data and adapt to changing systems Leveraging ML algorithms and Python programming, researchers and practitioners sophisticated spam detection able to accurately identify and filter unwanted emails - Develop the system.

This is a lie that could cause misuse of personal facts. Unsolicited mail messages also waste garage area and bandwidth. We use a shape of system mastering to save you such threats. It seems that system mastering algorithms are extra effective than consumer described rules due to the fact system gaining knowledge of algorithms are extra efficient, less difficult, and extra effective.

Junk or spam mail may be very risky in other approaches as nicely and may result in the leak of some sensitive information and a few viruses which include Trojans, worms, un-block able ads, Cryptocurrency miners and different .Malware. It is very critical to fight junk or spam mail due to the fact it could cause serious conditions. In different words, spam may be very demanding for customers. This paper aims to comprehensively evaluate the progress made in detecting spam emails through the implementation of machine techniques and Python programming.

In particular, it will explore the following areas:

- **Spam Email Filtering**

  In this section, we will dive into the world of spam email and discover the tell tale signs of these unwanted messages. We will also highlight the various methods spammers use to avoid detection. Additionally, we will define types of spam such as phishing schemes, unsolicited advertisements, and emails packed with

harmful attachments. Get ready to explore the dark side of your inbox.

- **Spam Detection Challenges**

  Despite technological advancements, spam detection is still a challenging task due to the dynamic nature of spam detection. This section will clarify the major challenges facing spam detection systems, including sophisticated spam techniques, data imbalance problems, and the need for real-time detection.

- **Machine Learning Techniques for Spam Detection**

  Machine learning provides a data-driven approach to spam detection, enabling systems to identify spam email indicator patterns This section will present the popular ML algorithms used in spam detection an overview of, e.g.

- **Supervised learning**

  Support Vector Machines (SVM), Naive Bayes, Decision Trees and other algorithms have been trained on labeled data sets to classify emails as spam or legitimate based on omitted features of content, sender information and metadata. \

- **Un-Supervised learning**

  Through powerful unsupervised learning techniques such as K-means clustering and anomaly detection, patterns and anomalies can be uncovered without the need for labeled training data. This proves essential in detecting emerging spam patterns or zero-day attacks that may have previously gone undetected.

- **Deep learning**

  Deep learning models, including Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), excel at learning complex lessons from unstructured email data, such as text , image These models enhance the representation of objects and the actual removal of layers is an improvement.

- **Using Python**
  Python, with its rich libraries and tools, is the preferred platform for ML-based spam detection systems This section will cover the use of Python libraries such as scikit-learn, Tensor Flow, and K-eras role for data pre-processing, model narrative training, and analysis.

- **Evaluation Metrics**

  Assessing the performance of unsolicited mail detection systems necessitates using suitable assessment metrics. This segment will elucidate generally used metrics including accuracy, precision, don't forget, F1-score, and ROC-AUC, elucidating their importance in gauging the effectiveness of ML-based totally spam detection models.

- **Case Study and Applications**

  In this section, we will explore real-world examples of using ML techniques for detecting spam. By looking at a variety of case studies and applications, we will gain a deep understanding of the practicality and effectiveness of ML-based spam detection systems in different industries and contexts.

- **Future Direction and Challenges**

The paper will conclude with the aid of discussing rising developments, future directions, and unresolved challenges in spam detection the usage of ML strategies and Python programming. Areas for in addition studies, such as ensemble techniques

II. LITERATURE SURVEY

Do not forget a situation where you regularly get hold of unsolicited mail messages at the internet when advertising and marketing or shopping your merchandise. That is a terrible marketing strategy and rip-off. As recipients, we don't have any manipulate over this unsolicited mail. It additionally consumes plenty of memory and precious time. Therefore, there is a exceptional need for a few strategies to reduce this kind of spam. In this paper, we advocate a machine learning-based totally spam detection mechanism. This example uses statistics from about 6,000 emails. This method will save you a variety of time and reminiscence. This statistics can be used to extract features that play an critical position in figuring out the correctness, accuracy, computational energy, and misclassification of the algorithm. Email systems are one of the most common .The organization's popular communication system. Empowering people from all over the world to identify spam emails work from the author of simultaneous letters and spam letters by ; It was discussed here Table 1 shows the comparison . Displaying the author's works with clear classification .Feature extraction techniques, datasets and approaches and disadvantages.

Paper [1] offers a machine gaining knowledge of model using facts from 6000 valid and invalid emails. This newsletter covers the stairs of making a dictionary, creating capabilities, building a gadget gaining knowledge of model, and comparing the gadget mastering version to create a spam filter version. The version is initialized by way of developing a dictionary that includes a library called stop phrases and removes all helper words.
After growing the dictionary, we processed the characters and checked the accuracy of the extracted functions. They use and check the Naive Bayes set of rules (probabilistic classifier) at

some point of the machine mastering model layout segment. This version has an application that offers junk or spam mail filtering for email sending. The filter out does no longer can help you see the gadgets inside the photograph, so the photo transfer option is disabled. Run backend structures and set up spam reporting modules to classify emails as unsolicited mail/non-unsolicited mail. This statistics can be in addition analyzed to identify non-compliant and compromised structures primarily based on crook proceedings. The server also shops person records along with username and timestamp. Target precise messages that are not spam because some messages are unsolicited mail in a single corporation however not spam in any other enterprise. This model takes emails as input after the usage of Naive Bayes set of rules and the output is classified as zero for no spam and 1 for junk or spam mail. Using this algorithm, an accuracy of 87.82% was achieved.

In Paper[2], spam messages were labeled the usage of a logistic regression version. It's miles a device studying classification set of rules. We use this model to are expecting the final results of a specific based variable. This logistic version can are expecting the likelihood of two sorts of responses: raw/spam. The spam folder includes 5,572 emails, together with ordinary mail and spam. The facts is divided into groups: a schooling organization and a testing group. 80% of the emails are used for education and 20% for checking out, and the version output is represented using 0s and 1s. Using the logistic regression set of rules, the accuracy reached 96.5%. There are various techniques for the classification of emails into spam and non-spam emails.

The information within the paper [3] carries about 5000 emails obtained via junk or spam mail Killer, which might be read by means of the Python bundle "Pyzmail". During pre-studying, pattern emails have been extracted and converted to standard textual content for content analysis [4]. This model offers two capabilities, that are restricted to the use of N-grams, the maximum typically used of depend vectorization, and tf-IDF (time-frequency converted frequency facts). N-grams and tf-IDF are generated by means of trying to find textual content patterns the use of content context. The maximum typically used count number vectorization is primarily based at the quantity of words that seem regularly in electronic mail content. Pipelines are created to offer records about the technique, making it less complicated to evaluate results. This pipeline includes three algorithms: Naive Bayes, Logistic Regression, and help Vector device.

The assessment manner is supported through precision, precision, remember, and F1 score. For fashions using feature set 1 (i.e. N-gram and tf-IDF), logistic regression achieved the exceptional accuracy of ninety eight.33%. In the instance the use of feature set 2 (i.e. vectorized phrase matter by using default), logistic regression achieves the best accuracy (99.33%). your phrases. This facts is split into components: training information and trying out statistics. [3]. In this example, unsolicited mail words are dealt with as tokens and every phrase is assigned a completely unique wide variety referred to as "tokenization". Count number Vectorizer. In shape() is a technique to examine and perceive emails the usage of system gaining knowledge of. There is paintings to do and a countdown. Temporal frequency (tf) offers the quantity of occurrences of a word in a specific record, while inverse report frequency (IDF) reduces the importance of a feature. TFIDF does no longer do away with touchy content. Stemming is executed to convert words into similar structures. Evaluation of a classifier is generally based totally on metrics together with accuracy, precision, remember, and F1 score. Comparing the effects obtained, we observed that combining the goods ended in better speed measurement accuracy of 98% [4].

This is presents a machine learning approach using Bayes's theorem and the Naive Bayes Classifier to help in the effective identification and blocking of spam emails. It will track the sender's IP address and comprise some user-friendly features like user management, compose mail, and voice messaging. It has used the datasets available from Kaggle and employed NLTK and TF-IDF for better accuracy in spam and non-spam differentiation. The proposed model has comparatively better results in detecting spam. The system provides a realistic solution to minimize unwanted emails and increase security; hence, in the future, it can be enhanced by adding more algorithms and features [5].

Spam is unsolicited email. The biggest challenge with spam filtering is spam filters. Classify real emails as spam or filter real spam emails as real. Spam and email are today, classification is a common problem for email servers, network performance, data security, and security. Information on the amount of spam is frequently and constantly increasing [7, 9]. Challenges are various developments how to catch and tag spam emails before they reach your users' inboxes. There many techniques and methods found in the literature survey to used for spam or junk filtering or detection. These methods can be devided into following five main categories [9].

- Content-based filtering techniques such as support vector machines (SVM), K-nearest neighbours, and neural .Naïve Bayes and networks. This approach creates an automated filter rule base for email classification [4].

- Item-based spam filtering is one of the most common approaches to spam filtering, also known as instance based . Extracts all spam and spam emails for all users using unobtrusive collection methods. Crossing Through the process of grouping and evaluation, all data are classified into two vectors for spam detection and spam detection non - spam email [9].

- Adaptive Spam Filtering Techniques: This approach identifies and checks spam emails by grouping them, When divided into groups, incoming emails are compared to each group. Determining whether an email is spam. Whether or not spam is predicted based on percentage similarity to the group the email belongs to. [8]

- Previous Similarity-Based Spam Filtering Techniques: This approach can be considered sample based and memory- based . The filtering method of machine learning techniques used, such as k-nearest neighbour (KNN) filters and classification Received training emails for a specific instance [9].

- Rules-based spam filtering techniques such as Spam Assassin. This approach uses predefined or heuristic rules. Evaluate a large number of patterns, usually regular expressions, against a selected message. Some Similar patterns will increase the rank of your message. Some ranking rules remain constant over time, while others do Continuous updates are required to ensure effective filtering [9].

III. METHODLOGY

A. Information series or Data Collection

Inside the first step, we downloaded the dataset together with two rows of 5572 statistics from Kaggle (https://www.kaggle.com/datasets/shantanudhakadd/e mail-unsolicited mail detection dataset-classification). So words and categories. We need to import the Python libraries required for it to work, along with NumPy, Pandas and sklearn, after which the downloaded file desires to be placed inside the 'read_csv' direction within the pandas library. Records pre-processing should be accomplished by checking for null values inside the statistics.

```
In [6]: raw_mail_data = pd.read_csv('C:\\Users\\91707\\Downloads\\mail_data.csv')

In [7]: print(raw_mail_data)

     Category                                            Message
0         ham  Go until jurong point, crazy.. Available only ...
1         ham                      Ok lar... Joking wif u oni...
2        spam  Free entry in 2 a wkly comp to win FA Cup fina...
3         ham  U dun say so early hor... U c already then say...
4         ham  Nah I don't think he goes to usf, he lives aro...
...       ...                                                ...
```

Fig.2 Information series or Data Collection

B. Label Coding

Next, label coding, described as the method of converting labels into numerical values inside the system reader, wishes to be accomplished. Junk or Spam mail messages are marked as "0" and Ham messages are marked as "1".

```
In [11]: mail_data.loc[mail_data['Category'] == 'spam', 'Category',] = 0
         mail_data.loc[mail_data['Category'] == 'ham', 'Category',] = 1
```

Fig.3 . Label coding

B. Feature Extraction

Now the information in the spam document is divided into training data and check information in the ratio of 8:2 after which feature extraction is done the use of tf-idf vectorizer which inverts the identification time frequency facts frequency. this is generally an set of rules used to convert text right into a meaningful numerical representation that may be used to healthy system algorithms to make predictions. TF -IDF Vectorizer can measure the originality of a word by using evaluating the range of instances a word appears in a record with the number of files in which that word appears.

```
In [52]: feature_extraction = TfidfVectorizer(min_df = 1, stop_words='english', lowercase=Tr

         X_train_features = feature_extraction.fit_transform(X_train)
         X_test_features = feature_extraction.transform(X_test)

         # convert Y_train and Y_test values as integers

         Y_train = Y_train.astype('int')
         Y_test = Y_test.astype('int')
```

Fig. 4 Feature Extraction

D. Model Schooling

In this model, we undertake logistic regression classifier to expect junk mail or spam emails. Logistic regression is one of the most famous gadget mastering algorithms and is a supervised machine studying algorithm. It's far used to expect a selected variable the use of the technique of impartial variables. The cost need to be a discrete or express fee. Generally "yes" or "no", zero or 1, authentic or fake, and so on. works, however gives values among zero and 1 in preference to giving absolute values among 0 and 1.

E. Version Checking Out or Model EVALUATING

Version testing is the manner of the usage of distinctive metrics to understand the overall performance, strengths and weaknesses of device mastering. Model assessment is important to evaluate the effectiveness of the version inside the early tiers of research and additionally plays a function in preserving the version.

F. Outcomes or Result

The comparison of four models—Logistic Regression, Naive Bayes, SVM, and Random Forest—reveals the different accuracies in spam detection. Logistic Regression and SVM stand out as the best among the four, where the train and test accuracy stands at about 98%. Naive Bayes is a bit less accurate, with about 96% for both the train and test data. The random forest is performing extremely well on the train data with 99% but relatively less on test data with 97%. This kind of behavior is a hint at overfitting. Overall, the results prove that Logistic Regression and SVM are the most reliable to carry out the spam classification task, providing a good balance between high accuracy and good generalization ability. These results demonstrate the importance of selecting the right model for optimal spam detection performance.

IV. PSEUDO CODE OF METHODLOGY

There are following pseudo-code for the spam mail detection based on Machine Learning :

A. Load necessary libraries

- Numpy
- Pandas
- sklearn (for train_test_split, Tfidf Vectorizer, Logistic Regression, Multi nomial NB, SVC, Random Forest Classifier, accuracy_score)
- matplotlib.pyplot

B. Load the dataset from the CSV file into a pandas DataFrame

- raw_mail_data = read_csv('file_path')

C. Replace null values in the dataset with empty strings

- mail_data = raw_mail_data.where(pd.notnull(raw_mail_data), '')

D. Convert 'Category' column values

- If 'Category' is 'spam', set it to 0
- If 'Category' is 'ham', set it to 1

E. Define features and labels

- X = mail_data['Message']
- Y = mail_data['Category']

F. Split the data into training and testing sets

- X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=3)

G. Extract features using TF-IDF vectorizer

- Initialize Tfidf Vectorizer with stop words set to 'english' and lowercase=True
- Fit and transform X_train to get X_train_features
- Transform X_test to get X_test_features

H. Convert Y_train and Y_test to integer type

I. Train and evaluate models

- Logistic Regression
- Train the model using X_train_features and Y_train
- Predict and calculate accuracy on training and test data
- Naive Bayes:
- Train the model using X_train_features and Y_train
- Predict and calculate accuracy on training and test data
- SVM:
- Train the model using X_train_features and Y_train
- Predict and calculate accuracy on training and test data
- Random Forest:
- Train the model using X_train_features and Y_train
- Predict and calculate accuracy on training and test data

J. Visualize the accuracies

- Create bar plots for training and test accuracies of each model

K. Plot the number of spam and ham emails

- Count the number of spam and ham emails
- Create a bar plot showing these counts

L. Example input mail

- Transform the input mail using the trained TF-IDF vectorizer
- Predict using the trained Logistic Regression model
- Print whether the mail is 'Spam' or 'Ham'

V. RESULT

In this study, logistic regression, naive Bayes, SVM, and random forest—four machine learning models—were compared for email spam detection. After preprocessing the data and converting the labels to binary, we split the dataset and applied TF-IDF vectorization. Each model was trained and tested for accuracy. Logistic Regression had 0.965 test accuracy, while Naive Bayes had 0.972, SVM 0.980, and Random Forest 0.978. SVM mostly outperforms the others, showing its robustness in high-dimensional spaces. Naive Bayes also performed very well. The highest accuracy test of SVM indicates its suitability for real-world spam filtering applications.

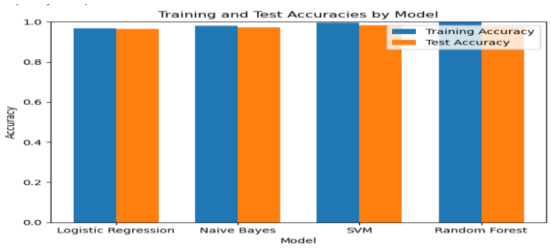

Fig.5 Comparison of all algorithms

VI. *CONCLUSION*

Via the over comes approximately, we'll conclude that the Naïve Bayes classifier beats all different classifiers. In show scenarios, spam the e-mails are growing.

Rapidly. We'd like remote better; a much better; a higher; a stronger;an progressed" a better display to apprehend unsolicited mail the e-mails to deal with that situation. Our proposed reveal witnesses the naïve Bayes classifier, which

Presents the probabilistic measurements that distinguish whether or not is junk mail or spam mail. Our proposed demonstrate accomplishes a cruel of 95 percent precision.

There is a huge scope for betterment in our project. The further enhancements can be made as follows:

"Spam filtering can be done based on the trusted and authenticated domain names."

"Spam e-mail classification has a significant role in classifying the e-mails and to differentiate the e-mails that are spam or non-spam."

"This approach may be applied with the large body in a position to distinguish good mails that are only the e mails they want to receive."

REFERENCES

[1] Nikhil Govil, Kunal Agarwal, Ashi Bansal, Astha Varshney. "A Machine Learning based Spam Detection Mechanism", IEEE Xplore Part Number: CFP20K25-ART; ISBN:978-1-7281-4889-2.

[2] Manoj Sethi, Sumesha Chandra, Vinayak Chaudhary, Yash, "Email Spam Detection using Machine Learning", International Research Journal of Engineering and Technology (IRJET).

[3] Jyoti Dake, Gunjan Memane, Prerana Katake, Samina Mulani "Email Spam Detection and Prevention using Machine Learning", International Journal of Advanced Research in Computer and Communication Engineering.

[4] Deepika Mallampati, K.Chandra Shekar and K.Ravikanth "Supervised Machine Learning Classifier for Email Spam Filtering", © Springer Nature Singapore Pte Ltd. 2019 and Engineering, https://doi.org/10.1007/978-981-13-7082-341.

[5] W.A, Awad & S.M, ELseuofi. (2011). Machine Learning Methods for Spam E-Mail Classification.

[6] International Journal of Computer Science & Information Technology. 3. 10.5121/ijcsit.2011.3112.

[7] 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

[8] D. K. Renuka, T. Hamsapriya, M. R. Chakkaravarthi and P. L. Surya, "Spam Classification Based on Supervised Learning Using Machine Learning Techniques," 2011 International Conference on Process Automation, Control and Computing, Coimbatore, 2011, pp. 1-7, doi: 10.1109/PACC.2011.5979035.

[8]. A.S. Aski, and K. N. Sourati, "Proposed efficient algorithm to filter spam using machine learning techniques," Pacific Science Review A: Natural Science and Engineering, 18(2), 145–149. https://doi.org/10.1016/j.psra.2016.09.017

E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi, & E. O. Ajibuwa,

[9] "Machine learning for email spam filtering: review, approaches and open research problems," Heliyon, 5(6), e01802. https://doi.org/10.1016/j.heliyon.2019.e01802

[10] S. Ajaz, M. T. Nafis, and V. Sharma, "Spam Mail Detection Using Hybrid Secure Hash Based Naive Classifier", International Journal of Advanced Research in Computer Science, Vol.8, No.5, pp.11951199, 2017.

[11] .B. Tang, S. Kay, and H. He, "Toward optimal feature selection in naive Bayes for text categorization", IEEE Transactions on Knowledge and Data Engineering, Vol.28, No.9, pp.2508-2521, 2016.

[12] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial Naive Bayes for Text Categorization Revisited", In: Proc. of Australasian Joint Conference on Artificial Intelligence, Vol.3339, Cairns, Australia, pp. 488-499, 2004.

[13] S. Youn, and D. McLeod, "A comparative study for email classification", Advances and innovations in systems, computing sciences and software engineering, pp.387-391, Springer, Dordrecht, 2007.

[14] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, G. Paliouras, and C. D. Spyropoulos, "An evaluation of naive bayesian anti-spam filtering", In: Proc. of the workshop on Machine Learning in the New Information Age, Barcelona, Spain, pp.9-17, 2000