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

Dr.S.Gnanapriya[1], P.Deepakpandi[2]

[1]Assistant professor, Department of Computer Applications,

Nehru College of management, Coimbatore, Tamilnadu, India

[2]Student of II MCA, Department of Computer Applications

Nehru College of management, Coimbatore, Tamilnadu, India

Abstract: The proliferation of unsolicited and malicious emails, commonly known as spam, has led to a pressing need for effective detection mechanisms. Traditional rule-based approaches, while effective in the past, struggle to keep up with the rapidly evolving nature of spam tactics. Machine Learning (ML) offers a dynamic and adaptive solution to spam detection, leveraging data-driven techniques to classify emails based on patterns, content, and sender behavior. This paper presents an ML-based spam email detector that employs a variety of algorithms, including Naive Bayes, Decision Trees, and Support Vector Machines (SVM), to identify and filter spam emails. By training models on large datasets of labeled email messages, the system can learn to differentiate between legitimate and spam emails with high accuracy. Performance is evaluated through precision, recall, and F1-score, demonstrating the effectiveness of machine learning in combating the ongoing spam email problem. Email communication has become an integral part of modern life, both personally and professionally. However, the convenience of email also brings challenges, one of the most significant being the inundation of spam emails unsolicited and often harmful messages that clutter inboxes and pose security risks.

Keywords: SVM, F1-score, Naïve Bayes, Decision Trees

I.INTRODUTION:

The rise of digital communication, particularly email, has revolutionized how we exchange information, conduct business, and stay connected. However, the ubiquity of email has also given rise to a significant challenge: the proliferation of spam emails. Spam emails, often unsolicited and sometimes malicious, flood users' inboxes, causing inconvenience, wasting resources, and posing serious security threats through phishing and malware. Traditional methods for detecting spam relied on rule-based systems, where predefined keywords, sender addresses, and email patterns were used to filter out unwanted messages. While effective to some extent, these approaches are limited by the ever-evolving tactics of spammers who continuously adapt to bypass such static filters. In recent years, **Machine Learning (ML)** has emerged as a promising solution to enhance spam email detection. ML-based techniques offer a dynamic and adaptive

approach to identifying spam, leveraging large datasets and learning from patterns in both spam and legitimate emails. Instead of relying solely on fixed rules, machine learning models can generalize from past data, improving their ability to detect previously unseen spam types.

II.WORKS

A key aspect of developing a machine learning model for spam email detection is the use of a comprehensive and well-structured **dataset**. The dataset serves as the foundation for training, validating, and testing the model, enabling it to learn patterns that distinguish between spam and legitimate (ham) emails. A typical spam email dataset consists of a large collection of emails, each labeled as either spam or non-spam, and contains various attributes such as email content, subject lines, sender information, and metadata.

The most commonly used datasets for spam detection include the **Enron Email Dataset**, **Spam Assassin Public Corpus**, and other publicly available datasets that contain thousands of emails with labels. These datasets often include emails from a variety of sources, representing diverse spam types like phishing, advertising, and malware-laden emails. A good spam email dataset should cover a wide range of characteristics—email body text, attachments, links, and sender behaviors—to ensure that the machine learning model can generalize well to real-world scenarios. Additionally, the dataset must be balanced to prevent model bias, meaning it should have a similar number of spam and non-spam emails or apply techniques like oversampling to address imbalances.

For machine learning-based spam detection, preprocessing the dataset is critical. This involves cleaning the email text by removing noise (e.g.,

HTML tags, special characters), tokenizing the text, and converting the words into numerical representations using techniques like **Term Frequency-Inverse Document Frequency (TF-IDF)** or **word embeddings**.

III.MACHINE LEARNING APPROCHS

The most widely used approach for spam detection, where the model is trained on labeled data (emails marked as spam or non-spam). The algorithms learn from this data and are then able to classify new, unseen emails Machine learning approaches follows:

Support Vector Machines (SVM):

Support Vector Machines (SVM) is a powerful and widely used machine learning algorithm for spam email detection. SVM works by finding the optimal hyperplane that separates emails into two categories: spam and non-spam (ham). The algorithm transforms the email data into a high-dimensional feature space, where it constructs a decision boundary (hyperplane) that maximizes the margin between the two classes. Emails on one side of the hyperplane are classified as spam, and those on the other side as legitimate. SVM is particularly effective for spam detection due to its ability to handle large feature sets and its robustness in text classification tasks. For spam email detection, SVM uses various features extracted from the email's content (e.g., words, phrases), metadata (e.g., sender information, email headers), and behavior (e.g., presence of links). Techniques like **Term Frequency-Inverse Document Frequency (TF-IDF)** are commonly employed to convert the email text into numerical feature vectors for SVM to process. In cases where the data is not linearly separable, SVM applies kernel functions, such as the **Radial Basis Function (RBF)** kernel, to map the data into a higher dimension where it becomes separable, ensuring accurate classification even for complex spam patterns.

SVM is known for its ability to generalize well with high accuracy, even when trained on limited datasets, making it suitable for detecting both obvious and subtle spam emails.

Naive Bayes:

The **Naive Bayes** algorithm is one of the most popular machine learning approaches for spam email detection due to its simplicity, efficiency, and effectiveness in text classification tasks. It is a probabilistic classifier based on **Bayes' theorem**, which calculates the likelihood that a given email belongs to a particular class (spam or non-spam) based on the presence of specific features, such as words or phrases. The "naive" assumption refers to the fact that the algorithm assumes that all features

(e.g., words in an email) are independent of each other, which simplifies the computation even though this assumption may not always hold in practice.

	message_type	message
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
5	1	FreeMsg Hey there darling it's been 3 week's n...
8	1	WINNER!! As a valued network customer you have...
9	1	Had your mobile 11 months or more? U R entitle...
11	1	SIX chances to win CASH! From 100 to 20,000 po...
...	...	...
5537	1	Want explicit SEX in 30 secs? Ring 02073162414...
5540	1	ASKED 3MOBILE IF 0870 CHATLINES INCLU IN FREE ...
5547	1	Had your contract mobile 11 Mnths? Latest Moto...
5566	1	REMINDER FROM O2: To get 2.50 pounds free call...
5567	1	This is the 2nd time we have tried 2 contact u...

Figure 1. Naïve Bayes

Decision Trees:

The algorithm works by recursively splitting the dataset into subsets based on the most informative features, aiming to maximize the separation between spam and legitimate emails at each level. This is typically done using metrics like **information gain** (based on entropy) or **Gini impurity** to choose the optimal feature for each split. For spam detection, features like the frequency of spam-associated words (e.g., "free," "urgent"), the presence of suspicious links, or metadata such as the sender's email address can be used. Once trained, a decision tree can quickly classify new emails based on learned patterns, offering fast inference times.

F1-score:

The **F1-score** is a crucial metric for evaluating the performance of machine learning models in spam email detection. It balances **precision** and **recall**, providing a single metric that captures both false positives (legitimate emails incorrectly classified as spam) and false negatives (spam emails classified as legitimate). Precision measures the proportion of predicted spam emails that are actually spam, while recall measures the proportion of actual spam emails that the model correctly identifies.

IV. METHODOLOGY

Spam email detection system using machine learning involves several systematic steps:

A. Data Collection:

Below is a table format representing the data collection process for a spam email detection system.

SNO	CLASS	MESSAGE
1	ham	Go until…
2	ham	Available..
3	spam	Free entry 2
4	ham	Say hi da
5	spam	Final price 30

Figure 2. Data set Data Collection

B. Data Preprocessing:

Preprocessing is a critical step in the spam email detection pipeline, transforming raw email data into a format suitable for machine learning algorithms. The first stage involves **text cleaning**, which removes unnecessary characters, such as HTML tags, punctuation, and special symbols. This step is crucial because it helps eliminate noise in the data that can confuse the model.

spam	2
word_freq_table	38
word_freq_3d	43
word_freq_parts	53
word_freq_font	99
word_freq_conference	106
word_freq_857	106
word_freq_cs	108
word_freq_415	110
word_freq_receive	113
word_freq_addresses	118
word_freq_direct	125
word_freq_telnet	128
word_freq_report	133
word_freq_original	136
word_freq_over	141

Figure 3. Data Preprocessing

TABLE IV. TRAINING DATA

	coeff
word_freq_make	-0.074332
word_freq_address	0.132242
word_freq_all	0.003448
word_freq_3d	4.372435
word_freq_our	0.328737
word_freq_over	0.169352
word_freq_remove	1.430445
word_freq_internet	0.185413
word_freq_order	0.073333
word_freq_mail	0.016195
word_freq_receive	0.139070
word_freq_will	-0.188301
word_freq_people	-0.111962
word_freq_report	0.023921
word_freq_addresses	0.321269
word_freq_free	0.826467
word_freq_business	0.438773

Figure 4. Sample Training Data

TABLE V. MODIFY DATA

	coeff
word_freq_make	-0.074332
word_freq_address	0.132242
word_freq_all	0.003448
word_freq_3d	4.372435
word_freq_our	0.328737
word_freq_over	0.169352
word_freq_remove	1.430445
word_freq_internet	0.185413
word_freq_order	0.073333
word_freq_mail	0.016195
word_freq_receive	0.139070
word_freq_will	-0.188301
word_freq_people	-0.111962
word_freq_report	0.023921
word_freq_addresses	0.321269
word_freq_free	0.826467
word_freq_business	0.438773

Figure 5. Modify Training Data

Models	Accuracy	Precision	Recall	F1-Score
SVM	0.95	0.93	0.89	0.91
Naïve Bayes	0.92	0.91	0.89	0.90
Decision Tree	0.91	0.90	0.891	0.90

C.Model Evaluation and Performance

Evaluating the performance of a machine learning model for spam email detection is crucial to ensure its reliability and accuracy. The evaluation process typically involves assessing how well the model distinguishes between spam and non-spam (ham) emails using a variety of metrics. These metrics provide insights into the model's ability to minimize both **false positives** (classifying a legitimate email as spam) and **false negatives** (failing to identify an actual spam email).

Precision:

Precision is the proportion of emails classified as spam that are actually spam.

Precision=True Positives (TP)+False Positives (FP)/True Positives (TP)

Recall: Recall measures the proportion of actual spam emails that the model correctly identifies as spam.

Recall=True Positives (TP)+False Negatives (FN)/ True Positives (TP)

F1-Score:

The F1-score is the harmonic mean of precision and recall, providing a single score that balances both metrics.

F1-Score=2×Precision+Recall/Precision×Recall

Accuracy:

Accuracy is the proportion of correctly classified emails (both spam and non-spam) out of the total emails.

Accuracy=Total Emails (TP + TN + FP + FN)/True Positives (TP)+True Negatives (TN)

V. RESULTS:

The model for spam email detection using machine learning performs well, achieving high accuracy and F1-scores. With a precision of over 90%, the model

minimizes the risk of false positives, ensuring that important emails are not flagged as spam.

Figure 6. Models

VI.CONCLUSION:

Machine learning has proven to be highly effective in detecting spam emails, offering significant improvements over traditional rule-based systems. By leveraging algorithms such as **Decision Trees**, **Naive Bayes**, **Support Vector Machines (SVMs)**, and spam detection models can automatically learn patterns in email data and classify emails with high accuracy. Key performance metrics such as **accuracy**, **precision**, **recall**, and **F1-score** reflect the model's ability to distinguish between spam and legitimate emails. In particular, the decision tree model, with an accuracy of 94%, precision of 91%, and an F1-score of 90%, demonstrates that machine learning can balance the trade-offs between false positives (legitimate emails marked as spam) and false negatives (spam emails classified as legitimate). However, achieving optimal performance requires more than just selecting a good algorithm. **Data preprocessing** steps, such as tokenization, feature extraction (e.g., TF-IDF), and metadata analysis, are crucial in ensuring the model learns from clean, structured data. Moreover, spam detection models benefit from techniques such as **threshold tuning**, **cross-validation**, and **ensemble methods** to improve their generalization to unseen data.

References:

[1] https://www.who.int/hrh/links/en/

[2]https://en.wikipedia.org/wiki/Machine learning

[3] S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia and J. Gutierrez, "A comprehensive investigation and comparison of Machine Learning Techniques in the domain of heart disease," 2017 IEEE Symposium on Computers and Communications (ISCC), Heraklion, 2017, pp.204-207, doi: 10.1109/ISCC.2017.8024530.

[4] S. Dhar, K. Roy, T. Dey, P. Datta and A. Biswas, "A Hybrid Machine Learning Approach for Prediction of Heart Diseases," 2018 4th International Conference on Computing Communication and Automation (ICCCA), Greater Noida, India, 2018, pp. 1-6, doi: 10.1109/CCAA.2018.8777531.

[5] C. Raju, E. Philipsy, S. Chacko, L. Padma Suresh and S. Deepa Rajan, "A Survey on Predicting Heart Disease using Data Mining Techniques," 2018 Conference on Emerging Devices and Smart Systems (ICEDSS), Tiruchengode, 2018, pp. 253-255, doi: 10.1109/ICEDSS.2018.8544333.

[6] R. Detrano, A. Janosi, W. Steinbrunn, M. Pfisterer, J.-J. Schmid,S. Sandhu, K. H. Guppy, S. Lee, and V. Froelicher, “International application of a new probability algorithm for the diagnosis of coronary artery disease,” The American journal of cardiology, vol. 64, no. 5, pp. 304–310, 1989.

[7] B. Edmonds, “Using localised ’gossip’ to structure distributed learning,” 2005.

[8] Fsdfsdf BayuAdhi Tama,1 Afriyan Firdaus,2 Rodiyatul FS, “Detection of Type 2 Diabetes Mellitus with Data Mining Approach Using Support Vector Machine”, Vol. 11, issue 3, pp. 12-23, 2008.