IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Machine Learning For Spam Email Detection

Ms. Shatabdi Mohanty and Ms. Sravani Jena

DEPARTMENT OF COMPUTER SCIENCE

CHAITANYA DEEMED TO BE UNIVERSITY,

MOINABAD, RANGA REDDY- 500075

Abstract: Spam emails have become a big problem nowadays, with the rapid growth of internet users around the globe. Spams are one of the main issues in the digital world. Spams not only influence the organizations financially but also bother individual email users. Spam emails are the most common problem on the internet. It's easy for spammers to send emails containing spam messages. Spammers can also steal important information from our devices, like files and contacts. Recently, various deep learning-based methods for word embedding approaches have been developed. These developments in the area of word representation could provide a feasible solution to such issues. In this research, we'll look into the effectiveness of machine learning techniques like Naïve Bayes and Support Vector Machine, as well as Natural Language Processing (NLP), for detecting spam emails.

Keywords: Machine Learning, Naïve Bayes, Support Vector Machine-nearest neighbor, Natural Language Processing (NLP), LSTM (Long Short-Term Memory).

I. INTRODUCTION

Spam email is the real problem on the internet. It refers to unsolicited messages that are more likely sent to a bunch of people for advertising products or services; even without the permission, we receive such spam mails. Despite our best efforts, spam is on the rise, and it's not just a minor exasperation. It jams our inboxes, wastes time, and slows down our communication. Once email filters were a reliable solution, but spammers are getting clever and can easily overcome traditional spam filters. Blocking emails manually from specific addresses isn't as effective as it once was. So now we are shifting towards machine learning and the knowledge engineering approach. Machine learning is better than knowledge engineering, as it does not follow a set of rules. Moreover, it uses training and test data. Training data contains mail, which is categorized as spam or ham. Natural Language Processing plays a significant role in detecting whether the email is unsolicited or not. NLP converts unstructured text into structured text and does text analysis for a better filtration of mail. NLP is used to enhance the accuracy of the model. Some of the machine learning techniques being used include content analysis, whitelists and blacklists, and community feedback. Naive Bayes is mostly suitable for improper content; it is successfully used in text filtering, analysis, classification, and recommender systems. The method is based on Bayes' theorem and is grounded on probabilistic reasoning. It uses conditional probability concepts to calculate the likelihood of a certain class given a collection of input characteristics. Given the naive assumption that naive Bayes makes, it is best suited for cases where the characteristics are conditionally independent of one another. The Naive Bayes algorithm adopts a simple and natural methodology. It determines the class probabilities and the

probability of the features given each class using a training dataset containing cases that have been labelled. Naive Bayes calculates the posterior probabilities for each class and places the instance in the class with the greatest probability when it is given a fresh, unlabeled instance. The computational effectiveness of Naive Bayes is one of its main benefits. It functions effectively even when the number of features is much more than the number of training examples and can handle huge datasets with high-dimensional feature spaces [2]. Because of this, Naive Bayes is especially well suited for applications that need to be real-time or have limited processing resources. Naive Bayes is extremely faster than the remaining methods to solve the problems.

II. LITERATURE REVIEW

A comprehensive literature survey on spam detection reveals a success from traditional rule-based filtering techniques to modify machine learning (ML) and artificial intelligence (AI). In recent studies, a thorough analysis of emails was conducted, focusing on header metadata, the body content using NLP techniques, and inspection of attachments to identify malicious structures. Classical ML techniques such as Naive Bayes, support vector machines (SVM), decision trees, random forests, and k-NN have been widely explored, but recent work trends on deep learning approaches capabilities of detecting sophisticated and evolving spam tactics. The authors fine-tuned a BERT model for email classification, achieving superior performance compared to a baseline deep neural network (DNN) that utilized Dense and Bi-LSTM layers, as well as recent classification methods like k-NN and Naive Bayes. This approach resulted in a rapid increase in accuracy to 98.67% and an F1-score of 98.66%. Remaining research extends spam detection beyond email; a deep learning-based framework was proposed for detecting Twitter spammers using both content and user metadata. Moreover, it introduced a novel spam detection using the Horse Herd Optimization Algorithm, which outperforms such ML models as KNN-GWO, SVM, and MLP. The related work on the LSTM structure for sentiment analysis is documented in an article; however, challenges related to computational resources have limited attention to recent news sources. Additionally, the computation of news source reliability using AI-driven approaches aims to enhance spam detection accuracy, robustness, and adaptability in response to evolving spam tactics.

III. METHODOLOGY

A. Data preprocessing:

The first step in any data analysis or machine learning pipeline is data preprocessing. It involves cleaning, transforming, and arranging raw data to check if it is accurate, consistent, and ready for modeling.

Steps in Data Preprocessing:

- Data cleaning: This is a step in ML that involves finding and removing any missing, duplicate, or unimportant data.

- Data Integration: This is the process after data cleaning; it combines data from multiple sources into a single, unified view. Data integration involves cleaning and transforming the data, as well as resolving any inconsistencies or conflicts that may exist between the different sources. The goal of data integration is to make data more useful and meaningful for the purposes of analysis and decision-making.

- Data Transformation: This is an important step in the data analysis process that involves the conversion, cleaning, and organizing of data into accessible formats.

- Data Reduction: It is the process to reduce the size of the dataset while still preserving the most important information.

1. Stop words:

Natural language processing tasks often involve filtering out extremely common words that provide not much meaning to a sentence. They can be safely ignored without losing the sense of the sentence. For instance, when searching for a query like "How to make a veg cheese sandwich," the search engine will look for web pages that include the words "how," "to," "make," "a," "veg," "cheese," and "sandwich." The search engine aims to find pages that feature the terms "how," "to," and "a," rather than those that provide recipes for a veg cheese sandwich, since these three words are very commonly found in the English language. If these three words were eliminated or disregarded and the focus was shifted to retrieving pages that included the keywords "veg," "cheese," and "sandwich," it would yield more relevant results.

2.Tokenization:

"Tokenization refers to the method of dividing a manuscript stream into phrases, symbols, words, or any expressive components known as tokens." These tokens may include words, characters, sub-words, or entire sentences. It helps to guess the text by various models. As tokenization happens at the word level. The tokens are separated by whitespaces like "line break" or "space" or by "punctuation characters." These white spaces and punctuations may or may not be involved in the resulting lists of tokens.

3.Bag of words:

Bag of words is a method to get features from text documents. Further, these features can be used for training machine learning algorithms. Bag of words creates a vocabulary of all the unique words present in all the documents in the training dataset.

B. CLASSIC CLASSIFIERS

Classification is a method of analyzing data that identifies the models representing significant data categories. A classifier or model is developed to predict class labels, such as "Is a loan application risky or safe?"

Data classification involves two phases:

- a learning step (creating the classification model) and

- a classification step

1. NAIVE BAYES:

The Naïve Bayes classifier was applied in 998 to identify spam. The Naïve Bayes classifier algorithm serves as a method for supervised learning. This Bayesian classifier operates on dependent events and relies on the probability of future occurrences based on previously observed events. Naïve Bayes is based on Bayes' theorem, which assumes that features are independent of one another. The Naïve Bayes classifier can be applied to classify spam emails, where the probability of words plays a critical role. If a word appears frequently in spam but rarely in legitimate emails, that email is likely to be spam. The Naïve Bayes classifier algorithm has become one of the most effective methods for email filtering. To achieve this, the model is trained using the Naïve Bayes filter, which functions efficiently. Naïve Bayes consistently computes the probability of each class, and the class with the highest probability is selected as the output. Naïve Bayes often yields precise results. It is utilized in various domains, including spam filtering.

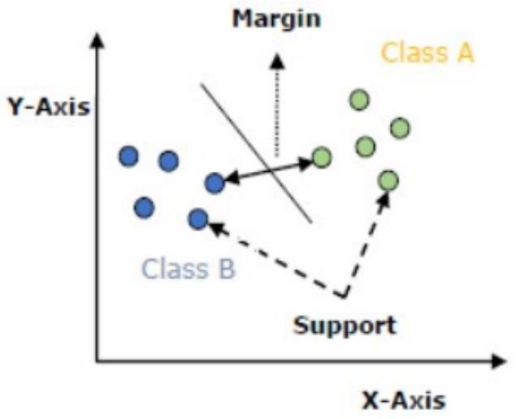
$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

$$P(B) = \sum_{y} P(B|A)P(A)$$

-(1)

2. SUPPORT VECTOR MACHINE

The Support Vector Machine (SVM) is a popular supervised learning algorithm. The Support Vector model is used for classification problems in machine learning techniques. Support Vector Machines are entirely based on the concept of decision boundaries. The main resolution of the Support Vector Machine algorithm is to create the line or decision boundary. The Support Vector Machine algorithm gives a hyperplane as an output, which classifies new samples In a two-dimensional space, a hyperplane is defined as a line that separates a plane into two sections, with each class located on one side.



3. DECISION TREE

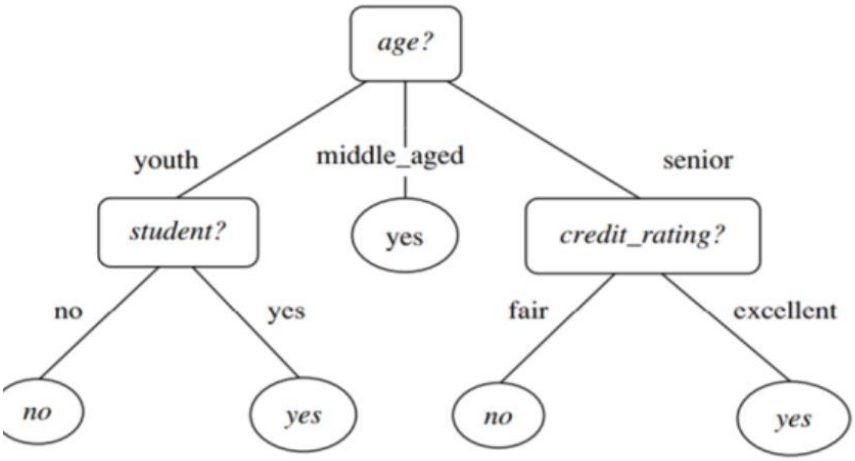
"Decision tree induction involves creating a decision tree from training tuples that are labeled by class." A decision tree is structured like a flowchart.

Internal node or non- leaf node= Test on attribute

Branch = shows outcome of the test

Leaf node= holds a class label

Top node is called root node.



Decision tree Induction:

The building of "decision tree classifiers" doesn't need "any domain knowledge or parameter setting that is suitable for examining knowledge." It handles multidimensional information The processes of learning and classifying in decision tree induction are straightforward and quick. Characteristic choice events are utilized to choose the characteristic that top-parcels the tuple into particular classes. At the point when the choice tree is manufactured, a significant number of the branches may result from or reflect commotion and anomalies in the preparation information. Tree pruning endeavors to recognize and evacuate such branches, with the objective of improving classifier precision on inconspicuous information.

Entropy using the frequency table of one attribute:(1)

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Entropy using the frequency table of two attributes:(2)

$$E(T,X) = \sum_{c \in X} P(c)E(c)$$

4. K-NEAREST NEIGHBOR

K-nearest neighbors is a supervised classification algorithm. This algorithm has some data points and data vectors that are separated into several classes to predict the classification of new sample points."

K-nearest neighbor is a LAZY algorithm. A 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 points based on a similarity measure that can be Euclidean distance. The Euclidean distance measure calculates Euclidean distance and identifies who its neighbors are.

$$dist((x, y), (a, b)) = \sqrt{(x - a)^2 + (y - b)^2} \quad -(5)$$

C. ENSEMBLE LEARNING METHODS

"Ensemble methods in machine learning are methods that take several base models to produce a predictive model in order to decrease variance by using bagging bias by using boosting predictions using stacking Two Types Sequential—this method involves creating base classifiers one after another. Parallel—here base classifiers are in parallel.

1. RANDOM FOREST CLASSIFIER

The random forest classifier is an ensemble tree classifier consisting of different types of decision trees that are of different shapes and sizes. The random sampling of the training data when building a tree. A random subgroup of input features when splitting at a node in a tree. If you have randomness, the randomization will make the decision tree look less correlated so that the generalization error (features of the tree should not look the same) of the ensemble can be improved.

2. BAGGING

A bagging classifier is an ensemble method that trains base classifiers on random portions of the original datasets, subsequently merging their predictions through voting or averaging to reach a final decision. Bagging combines the concepts of bootstrapping and aggregation.

Bagging= **B**ootstrap **AGG**regat**ING**

Bootstrapping helps to lessen the variance of the classifier, and it also declines the overfitting by just resampling the data from the training data with the same cardinality as in the original data set. High variance is not good for the model. Bagging is a very effective method for limited data, and by just using samples you are able to get an estimate by aggregating the scores.

BOOSTING AND ADABOOST CLASSIFIER

"Boosting is an ensemble technique employed to develop a robust classifier by combining several weak classifiers." Boosting is completed by creating a model from a training data set, then creating another model that will precisely identify the faults of the first model." [8] In the boosting model, they are added till the training set is predicted properly.

AdaBoost= **ADA**ptive **Boost**ing

AdaBoost is the first fruitful boosting algorithm that was settled for binary classification. The boosting is understood by using AdaBoost.

CONCLUSION

The performance of various algorithms was compared. Naïve Bayes achieved an accuracy of approximately 98%, SVM achieved 96%, and logistic regression achieved 94%. The Naïve Bayes model demonstrated superior generalization due to its probabilistic nature and efficiency in handling sparse data. Precision and recall metrics further validated the robustness of the approach, ensuring minimal false positives.

This research successfully demonstrates the application of machine learning algorithms for spam email detection. Among all tested models, the Naïve Bayes classifier proved to be the most effective. Future work may involve exploring deep learning models like LSTM or BERT for improved contextual understanding and accuracy. Integration with cloud-based systems can further enhance scalability and deployment efficiency.

REFERENCES

1. Almeida, T. A., et al. (2011). Contributions to the Study of SMS Spam Filtering: New Collection and Results. Proceedings of the 11th ACM Symposium on Document Engineering.

2. Guzella, T. S., & Caminhas, W. M. (2009). A Review of Machine Learning Approaches to Spam Filtering. Expert Systems with Applications, 36(7), 10206–10222.

3. Pedregosa, F., et al. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

4. McCallum, A., & Nigam, K. (1998). A Comparison of Event Models for Naïve Bayes Text Classification. AAAI Workshop on Learning for Text Categorization.