INTERNATIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT (IJSREM)
VOLUME: 08 ISSUE: 04 | APRIL - 2024 SJIF RATING: 8.448 ISSN: 2582-3930

Email Spam Detection using Naive Bayes Algorithm Optimized with Particle Swarm Optimization

S. Harsha Vardhan Reddy
Student of ECE,
RVR&JC CE,
Guntur, A.P India,
harsha28102@gmail.com

Y. Lakshmi Narasimha
Student of ECE,
RVR&JC CE,
Guntur, A.P India,
yanamalamandanarasimha@gmail.com

N. Bhanu Thahera
Student of ECE,
RVR&JC CE,
Guntur, A.P India,
bhanuthahera@gmail.com

V. Harsha
Student of ECE,
RVR&JC CE,
Guntur, A.P India,
vemulapalliharsha02@gmail.com

Abstract — Spamming through emails has always been one of the most exploited methods for the fraudsters. Spam emails are inappropriate and unwanted messages usually sent to breach security. Spam emails encompass various forms, including advertisements, commercial segments, as well as false promises of prices, discounts, or job opportunities. spammers continuously adapt their strategies to increase the chances of their emails bypassing filters and being opened by recipients. To overcome this issue we present a sophisticated approach for machine learning algorithm with Optimization to detect spam emails.A machine learning based Na¨ıve Bayes (NB) algorithm by using Particle Swarm Optimization (PSO). Naive Bayes is effective for email spam detection due to its ability to handle high-dimensional data (like text) and its simplicity, making it efficient even with large datasets. The performance of Na¨ıve Bayes algorithm is compared with the Support Vector Machine (SVM) using metrics such as accuracy, precision, recall, and F1-score.

Keywords—spam emails, Machine Learning, Naive Bayes, Particle Swarm Optimization, Support Vector Machine

I. INTRODUCTION

With the exponential growth of email communication, the issue of spam has become a significant problem, affecting individuals and organizations alike. Spam emails not only clutter inboxes but also pose security risks and waste valuable resources. To combat this issue, various spam filtering techniques have been developed, among which the Naive Bayes algorithm has emerged as a popular choice due to its simplicity and effectiveness. The Naive Bayes algorithm is a probabilistic classifier based on Bayes' theorem, which assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. Despite this "naive" assumption, Naive Bayes has shown remarkable performance in text classification tasks, making it particularly suitable for email spam detection. In this paper, we present a study on the application of the Naive Bayes algorithm to the task of email spam detection. We discuss the theoretical foundations of the algorithm, its implementation in the context of spam filtering, and its performance evaluation using a real-world email dataset. We also explore a strategy for optimizing the algorithm's performance by integrating Particle Swarm Optimization (PSO) with Naïve Bayes for feature selection in email spam detection. PSO is a successfully applied in various optimization problems, including feature selection.

In our study, we use PSO to optimize the feature selection process in Naive Bayes for email spam detection, aiming to improve the algorithm's performance by selecting the most relevant features and discarding irrelevant ones. We evaluate the effectiveness of this approach through experiments on a real-world email dataset, demonstrating its potential in enhancing the spam detection accuracy of the Naive Bayes algorithm to achieve better spam detection accuracy.

II. RELATED WORK

Cut down on the time and resources required to manually manage and sift through enormous volumes of spam emails to increase business productivity and efficiency. All things considered, spam email detection is an essential tool for bolstering security, improving user experience, and maximizing email system efficiency [1]. Methods such as deny listing and allow listing are only sometimes successful as fraudsters always search for new ways[2].
Various classification algorithms, including Support Vector Machines (SVM), Random Forest (RF), and K-Nearest Neighbours (KNN) on the available dataset. The evaluation is performed based on the metrics accuracy, precision and recall. The results showed that SVM had the highest accuracy and precision, while Random Forest(RF) had the highest memory[3,4]. Observed better results with preprocessing steps in comparison with results with without pre-processing[5,6]. Integrated approach improves the results in comparison with individual SVM and NB approaches[7]. when experimented on large datasets linear SVM performs better for text based spam classification when compared with non linear SVM model [8].

III. CONCEPTS

A. Naive Bayes Approach

The Naive Bayes algorithm is a simple yet powerful algorithm for classification based on Bayes' theorem with an assumption of independence among features. The Naive Bayes algorithm assumes that the presence of a particular feature in a class is independent of the presence of any other feature. This is a strong and often unrealistic assumption, but it simplifies the calculation and is computationally efficient. The algorithm calculates two probabilities: Prior probability

and Likelihood. To make a prediction, the algorithm calculates the probability of each class label for a given set of features using Bayes theorem and the Naive Bayes assumption, and selects the class label with the highest probability.

B. Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a concept based on swarm intelligence. The social behaviour of schools of fish and flying birds served as inspiration for the authors. PSOs operate using the property of stochastic distribution to first identify the local search solution. Each particle then shares its solution with the group to determine the global solution. We refer to this attribute as the global optimization property. By using iterations, this algorithm gets closer to the ideal answer. In the beginning, particles in the form of a population of N particle solution are introduced into the process by a careless fly. The location of the ith particle is represented as a point in the S-dimensional space, where S is the number of variables involved. In the entire process, particles try to find the global best solution. The PSO algorithm operates using two primary dynamic vectors that alter particle location and velocity in accordance with to how the many particles interact with one another as each particle serve as a remedy. Every particle has the capacity to alter their trajectory based on knowledge and property sharing with additional particles with each iteration to produce a better solution.

C. Support Vector Machine

Support Vector Machine (SVM) is a powerful supervised machine learning algorithm based on the concept of finding the hyperplane that best separates different classes in the feature space. Support vectors are the data points that lie closest to the decision boundary. These points are crucial in defining the hyperplane and are used to make predictions.

IV. PROPOSED METHOD

In this section, we present an integrated concept of using Naive Bayes (NB) and Particle Swarm Optimization (PSO) to detect spam emails. NB, with its probability distribution property, determines the possible class for email content (spam or non-spam) based on keywords in the email's textual data. PSO is then applied to optimize the parameters of the NB approach, enhancing accuracy, search space exploration, and the classification process. The workflow of our proposed concept is illustrated in Figure 1. To provide a clear understanding, we outline a step-by-step algorithm for processing an individual email:

1. Select Email: Choose a random email from the ling spam dataset for experimentation.
2. Pre-processing: Convert the raw email into a format suitable for feature extraction and classification by tokenizing, stemming, and removing stop words.
- Tokenization: Split the email into individual keywords.
- Stop Word Removal: Eliminate common, non-informative words from the tokens.
- Stemming: Reduce words to their root form for normalization.
3. Feature Selection: Apply CFS to select relevant features from the pre-processed data.

$$\text{Merit(F)} = \frac{k+k(k-1)*AvgCorr(F)}{k.AvgCorr(F,c)} \quad (4.1)$$

Where:

• F is the subset of features being evaluated.
• c is the target class variable.
• k is the number of features in subset F.
• AvgCorr(F, c) is the average feature-class correlation over all features in subset F.
• AvgCorr(F) is the average feature-feature correlation over all pairs of features in subset F.
4. Calculate Probability Distribution: Use NB to calculate the probability distribution of tokens along with selected features.

$$\text{P(y|x)} = \frac{P(x|y)P(y)}{P(x)} \quad 4.2$$

Where, x is any feature vector set (x1, x2, x3, . . . xn) and y are the class variables with m possible outcomes (y1, y2, y3, . . . yn). P(y/x) stands for posterior probability, P(x/y) is any particular class on which P(y/x) is dependent. P(x) is evidence depending on the known feature variables, P(y) is the prior probability. So, Naive Bayes classification model consists of set of probabilities of prior probability, class conditional probability and posterior probability.
5. Optimize Parameters: Apply PSO to optimize the parameters of the NB approach.
- Particle Initialization: Treat each token as a particle, which initially randomly explores the search space.
- Velocity Update: Adjust the velocity of particles based on their current position and the best positions found so far.
- Position Update: Update the position of particles based on their velocity, seeking the best solutions.

The velocity updates in PSO can be calculated using the formula given below by Equation (3):

$$V_{i(t+1)} = \omega V_{i(t)} + c_1 r_1 (P_{i(t)} - X_i(t)) + c_2 r_2 (P_g - X_{i(t)}) \quad .....(4.3)$$

Now, V_i is the new velocity. So, the position of the particle updates with the velocity as defined with Equation (4):

$$Xi(t+1) = Xi(t) + Vi(t+1) \quad . . . (4)$$

Update the positions for each particle and store the global best solutions.
6. Classify Tokens: Based on the feature similarity evaluated using PSO, classify tokens as spam or non-spam.

7. Final Classification: Evaluate the probability of spam or non-spam tokens in the sentence for the final classification.
8. Store and Repeat: Store the email as spam or non-spam and repeat the process for all emails.

This integrated approach enhances the accuracy of spam email detection by effectively combining the strengths of NB and PSO.

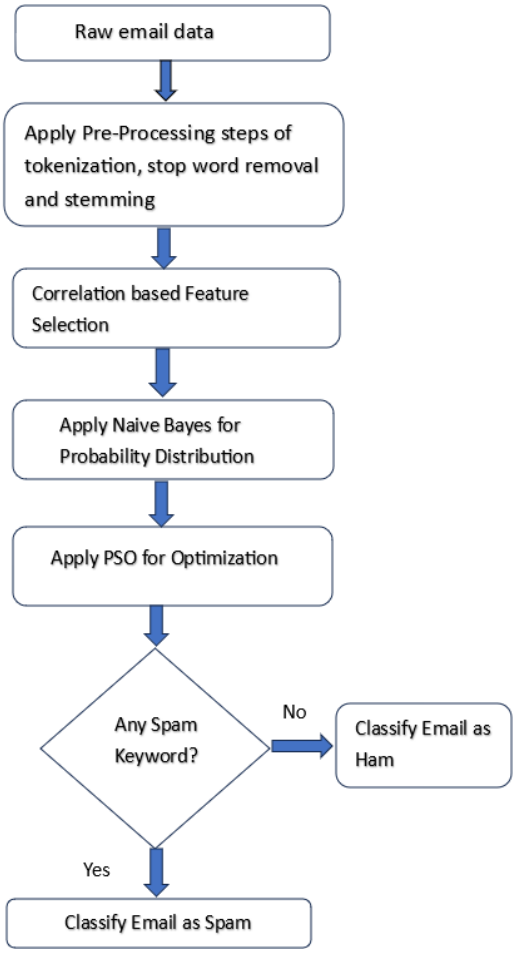

Fig: 1 Flow Chart of Email Spam detection

V. EXPERIMENTAL RESULTS

A. Dataset Used

The proposed method is experimented on the dataset from the ”kaggle” website. 5570 emails are selected randomly from this spam dataset. Out of these 5570 emails,3889 emails are used for training,1681 are used as testing emails for testing by maintaining a 70:30 ratio. Initially, training step is performed using NB and proposed integrated approach of NB & PSO. Then, based on the testing emails, results are evaluated for individual NB and proposed integrated approach of NB & PSO.

B. Evaluation Parameters

Performance of proposed algorithm is evaluated in terms of precision, recall, f-measure and classification accuracy. These parameters can be calculated with the help of True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). These measures are defines as below.
TP can be defined as the numbers of spam emails are correctly identified as spam.
TN can be defined as the numbers of non-spam emails are correctly identified as non-spam.
FP can be defined as the numbers of non-spam emails are incorrectly identified as spam.
FN can be defined as the numbers of spam emails are incorrectly identified as non-spam.

C. Results and Comparison

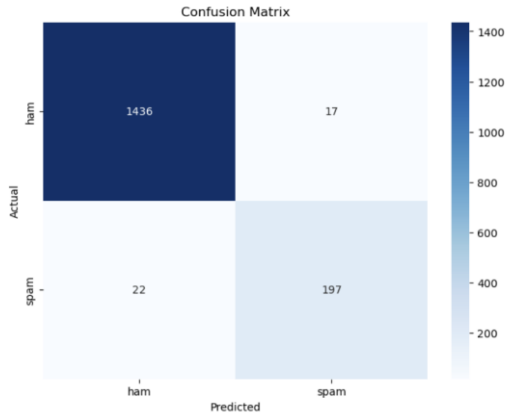

Fig: 2 Confusion Matrix

This exploration sheds light on its potential uses in improving antennas.

TABLE 1 Performance Metrics

Parameter	Formula	Value
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	0.985
Precision	$\frac{TP}{TP+FP}$	0.94
Recall	$\frac{TP}{TP+FN}$	0.95
F1 score	$2 * \frac{Precision * Recall}{Precision + Recall}$	0.94

Figures 2,3 indicates the comparision between non-uniform and uniform circular array’s radiation pattern.

The calculated values of TP, TN, FP and FN using individual NB and integrated proposed concept are obtained from the confusion matrix shown in Figure 2. Further, calculated values of precision, recall, f-measure and accuracy for integrated approach are shown in table I.

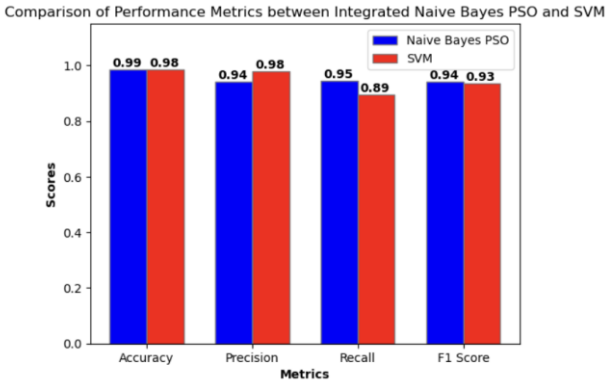

Fig: 3 Comparison Graph

The figure 3 shows the Comparison of Integrated concept and the Linear SVM model.The metrics are calculated considering the spam as a positive label. From the comparison graph, it can be seen that proposed integrated approach of NB and PSO has better results interms of accuracy, recall and f-measure when compared with with SVM.

VI. CONCLUSION

For email spam identification, we have employed an integrated strategy of NB and PSO and performed experimentation on the spam dataset from ”kaggle” website. From the evaluated results, it can be declared that the proposed integrated concept performed better in comparison with the standard Linear SVM model.

REFERENCES

[1] Towards Data Science, “How To Design A Spam Filtering System with Machine Learning Algorithm,” Dec. 2018.

[2] N. Ahmed, R. Amin, H. Aldabbas, D. Koundal, B. Alouffi, and T.Shah, “Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges,” Security and Communication Networks, vol. 2022. Hindawi Limited, 2022. doi: 10.1155/2022/1862888.

[3] jpinfotech, “Email Spam Detection Using Machine Learning Algorithms”,[Online]. Available: https://jpinfotech.org/email-spamdetectionusing-machine-learning-algorithms/

[4] M. Tope and M. E. Student, “Email Spam Detection using Naïve Bayes Classifier,” IJSDR1906001 International Journal of Scientific Development and Research (IJSDR) www.ijsdr.org, vol. 4, no. 6, Jun. 2019, [Online]. Available: www.ijsdr.org

[5] Tuteja, Simranjit Kaur, and Nagaraju Bogiri. ”Email Spam filtering using BPNN classification algorithm.” In Automatic Control and Dynamic Optimization Techniques (ICACDOT), International Conference on, pp.915-919. IEEE, 2016.

[6] Harisinghaney, Anirudh, Aman Dixit, Saurabh Gupta, and Anuja Arora.”Text and image based spam email classification using KNN, Na¨ıve Bayes and Reverse DBSCAN algorithm.” In Optimization, Reliabilty, and Information Technology (ICROIT), 2014 International Conferenceon, pp. 153-155. IEEE, 2014

[7] Feng, Weimiao, Jianguo Sun, Liguo Zhang, Cuiling Cao, and QingYang. ”A support vector machine based naive Bayes algorithm for spam filtering.” In Performance Computing and Communications Conference(IPCCC), 2016 IEEE 35th International, pp. 1-8. IEEE, 2016.