DETECTING SPAM EMAIL WITH MACHINE LEARNING OPTIMIZED WITH BIO INSPIRED METAHEURISTIC ALGORITHMS

Ms.A. Sowshna[1], P. Manisha[2], Manaal Ahmed Kabeer[3], K.Pratibha[4]

[1]Assistant Professor,
Department of Computer Science & Engineering,Sridevi Womens Engineering College,
Hyderabad, Telangana
[2,3,4]Undergraduate Student, Department of Computer Science & Engineering,
Sridevi Womens Engineering College,Hyderabad, Telangana

Abstract: Electronic mail has eased communication methods for many organisations as well as individuals. This method is exploited for fraudulent gain by spammers through sending unsolicited emails. This article aims to present a method for detection of spam emails with machine learning algorithms that are optimized with bio-inspired methods. A literature review is carried to explore the efficient methods applied on different datasets to achieve good results. An extensive research was done to implement machine learning models using PSO and BAT. Multi-Layer Perceptron on seven different email datasets, along with feature extraction and pre-processing. The bio-inspired algorithms like Particle Swarm Optimization and Genetic Algorithm were implemented to optimize the performance of classifiers. Multinomial Naïve Bayes with Genetic Algorithm performed the best overall. The comparison of our results with other machine learning and bio-inspired models to show the best suitable model is also discussed.

Keywords: Bio inspired algorithm, Particle Swarm Optimization algorithm BAT algorithm.

I.INTRODUCTION

Machine learning models have been utilized for multiple purposes in the field of computer science from resolving a network traffic issue to detecting a malware. Emails are used regularly by many people for communication and for socialising. Security breaches that compromises customer data allows 'spammers' to spoof a compromised email address to send illegitimate (spam) emails. This is also exploited to gain unauthorized access to their device by tricking the user into clicking the spam link within the spam email, that constitutes a phishing attack. Many tools and techniques are offered by companies in order to detect spam emails in a network. Organisations have set up filtering mechanisms to detect unsolicited emails by setting up rules and configuring the firewall settings. Google is one of the top companies that offers 99.9% success in detecting such emails. There are different areas for deploying the spam filters such as on the gateway (router), on the cloud hosted applications or on the user's computer. In order to overcome the detection problem of spam emails, methods such as content-based filtering,[1] rule-based filtering or Bayesian filtering have been applied. Unlike the 'knowledge engineering' where spam detection rules are set up and are in constant need of manual updating thus consuming time and resources, Machine learning makes it easier because it learns to recognise the unsolicited emails (spam) and legitimate emails (ham) automatically and then applies those learned instructions to unknown incoming emails[2]. The proposed spam detection to resolve the issue of the spam classification problem can be further experimented by feature selection or automated parameter selection for the models. This research conducts experiments involving five different machine learning models with Particle Swarm Optimization (PSO) and Genetic Algorithm (GA). This will be compared with the base models to conclude whether the proposed models have improved the performance with parameter tuning.

The Scope of this paper aims to achieve the following:

1) To explore machine learning algorithms for the spam detection problem.
2) To investigate the workings of the algorithms with the acquired datasets.
3) To implement the bio-inspired algorithms.
4) To test and compare the accuracy of base models with bio-inspired implementation.

2. RELATED STUDY

Naive Bayes classifiers are widely used to filter spam emails, however, the strong independence assumptions between features limit their performance in accurately identifying spams. To address this issue, we proposed a support machine vector based naive Bayes - SVM-NB - filtering system. The SVM-NB first constructs an optimal separating hyperplane that divides samples in the training set into two categories. For samples located nearby the hyperplane, if they are in different categories, one of them will be eliminated from the training set. In this way, the dependence between samples is reduced and the entire training sample space is simplified. With the trimmed training set, the naive Bayes algorithm is applied to classify emails in the test set. The SVM-NB system is evaluated with the dataset obtained from DATAMALL. Experiment results demonstrate that SVM-NB can achieve a higher spam-detection accuracy and a faster classification speed. upsurge in the volume of unwanted emails called spam has created an intense need for the development of more dependable and robust antispam filters. Machine learning methods of recent are being used to successfully detect and filter spam emails. We present a systematic review of some of the popular machine learning based email spam filtering approaches. Our review covers survey of the important concepts, attempts, efficiency, and the research trend in spam filtering. The preliminary discussion in the study background examines the applications of machine learning techniques to

the email spam filtering process of the leading internet service providers (ISPs) like Gmail, Yahoo and Outlook emails spam filters. Discussion on general email spam filtering process, and the various efforts by different researchers in combating spam through the use machine learning techniques was done. Our review compares the strengths and drawbacks of existing machine learning approaches and the open research problems in spam filtering. We recommended deep leaning and deep adversarial learning as the future techniques that can effectively handle the menace of spam emails.

The increasing volume of unsolicited bulk e-mail (also known as spam) has generated a need for reliableanti-spam filters. Machine learning techniques now days used to automatically filter the spam e-mail in avery successful rate. In this paper we review some of the most popular machine learning methods(Bayesian classification, k-NN, ANNs, SVMs, Artificial immune system and Rough sets) and of theirapplicability to the problem of spam Email classification. Descriptions of the algorithms are presented,and the comparison of their performance on the SpamAssassin spam corpus is presented.[4]

Email has become one of the fastest and most economical forms of communication. However, the increase of email users has resulted in the dramatic increase of spam emails during the past few years. As spammers always try to find a way to evade existing filters, new filters need to be developed to catch spam. Generally, the main tool for email filtering is based on text classification. A classifier then is a system that classifies incoming messages as spam or legitimate (ham) using classification methods. The most important methods of classification utilize machine learning techniques. There are a plethora of options when it comes to deciding how to add a machine learning component to a python email classification. This article describes an approach for spam filtering using Python where the interesting spam or ham words (spam-ham lexicon) are filtered first from the training dataset and then this lexicon is used to generate the training and testing tables that are used by variety of data mining algorithms. Our experimentation using one dataset reveals the affectivity of the Naïve Bayes and the SVM classifiers for spam filtering.

3. EXISTING SYSTEM

S. L. Marie-Sainte and N. Alalyani used the Firefly algorithm with SVM. The researchers experimented with the Arabic text with feature selection and also hybrid[5]. The paper concluded that the proposed method outperforms the SVM itself.

E. A. Natarajan, S. Subramanian, and K. Premalatha proposed Enhanced Cuckoo Search (ECS) for bloom filter optimization. This is where the weight of the spam word is considered. It was concluded that their proposed optimization technique of ECS outperforms the normal Cuckoo search.

DISADVANTAGES OF EXISTING SYSTEM

- Spam detection rules are set up and are in constant need of manual updating thus consuming time and resources.
- The problem of selecting the set of attributes is NP-hard.
- Less Accuracy.
- More time taking process.

4. PROPOSED SYSTEM:

This method is exploited for fraudulent gain by spammers through sending unsolicited emails. This article aims to present a method for detection of spam emails with machine learning algorithms that are optimized with bio-inspired methods. A literature review is carried to explore the efficient methods applied on different datasets to achieve good results. An extensive research was done to implement machine learning models using PSO and BAT.Multi-Layer Perceptron on seven different email datasets, along with feature extraction and pre-processing. The bio-inspired algorithms like Particle Swarm Optimization and Genetic Algorithm were implemented to optimize the performance of classifiers. Multinomial Naïve Bayes with Genetic Algorithm performed the best overall. The comparison of our results with other machine learning and bio-inspired models to show the best suitable model is also discussed[6].

ADVANTAGES OF PROPOSED SYETEM

The bio-inspired algorithms like Particle Swarm Optimization and Genetic Algorithm were implemented to optimize the performance of classifiers.

5. TEST CASES

Use case ID	Detecting Spam Email With Machine Learning Optimized With Bio-Inspired Metaheuristic Algorithms
Use case Name	Home button
Description	Display home page of application
Primary actor	User
Precondition	User must open application
Post condition	Display the Home Page of an application
Frequency of Use case	Many times
Alternative use case	N/A
Attachments	N/A

Table no1. Test cases.

6. ARCHITECTURE OF THE PROJECT

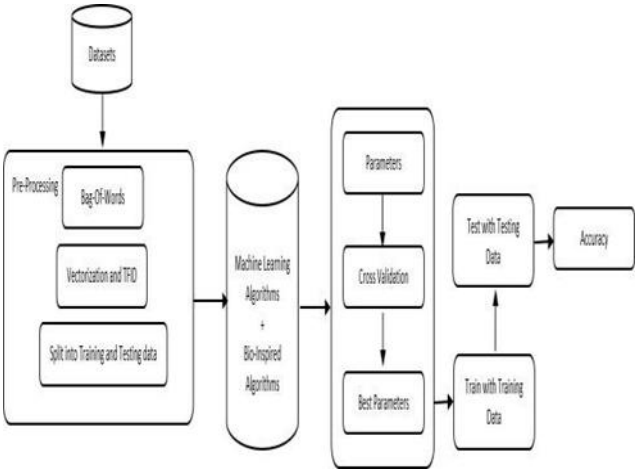

Fig1. Architecture

1) Upload Dataset: using this module we will upload dataset to application
2) Preprocess Dataset: using this module we will clean all dataset messages by removing stop words and special symbols
3) TFIDF feature engineering: using this module we will convert all words into vector where all words will be arrange as vector column names and the count of each word will be arrange as vector rows
4) Machine Learning Algorithms with PSO: using this module we will train all 5 above describe machine learning algorithms by using optimised features from PSO
5) Machine Learning Algorithms with BAT: using this module we will train all 5 above describe machine learning algorithms by using optimised features from BAT
6) Comparison Graph: will plot accuracy, precision, recall and FSCORE graph for all algorithms with PSO and BAT.

7. CONCLUSION AND FUTURE SCOPE

The project successfully implemented models combined with bio-inspired algorithms. The spam email corpus used within the project was both numerical as well as alphabetical. Approximately 50,000 emails were tested with the proposed models. Genetic Algorithm worked better overall for both text-based datasets and numerical-based datasets than PSO. The PSO worked well for Multinomial Naïve Bayes and Stochastic Gradient Descent, whereas GA worked well for Random Forest and Decision Tree. Naïve Bayes algorithm was proved to have been the best algorithm for spam detection[3]. This was concluded by evaluating the results for both numerical and alphabetical based dataset. The highest accuracy provided was 100% with GA optimization on randomized data distribution for 80:20 train and test split set on Spam Assassin dataset. In terms of F1-Score, precision and recall, Genetic Algorithm had more impact than PSO on MNB, SGD, RF and DT[8].

We plan to further carry out the machine learning algorithms to optimize and compare with different bio-inspired algorithms such as Firefly, Bee Colony and Ant Colony Optimization as researched in the previous sections. We could also explore the Deep learning Neural Network with PSO and GA by exploring different libraries such as Tensor Flow's DNN Classifier or similar. We found that the Neural Network algorithm could have worked better with more dimension like providing broader range of values for learning rate, activation, solver, and alpha. If this project is taken further, implementation for MLP could be done through Keras or Tensor Flow with GPU
application. This will allow the user to input other parameters and a range of possibilities as their key values.

8. REFERENCE

1. W. Feng, J. Sun, L. Zhang, C. Cao, and Q. Yang, ``A support vectormachine based Naive Bayes algorithm for spam ltering," in Proc. IEEE35th Int. Perform. Comput.Commun. Conf. (IPCCC), Dec. 2016.
2. E. G. Dada, J. S. Bassi, H. Chiroma, S. M. Abdulhamid, A. O. Adetunmbi,and O. E. Ajibuwa, ``Machine learning for email spam ltering: Review,approaches and open research problems," Heliyon, vol. 5, no. 6, Jun. 2019,Art. no. e01802, doi: 10.1016/j.heliyon.2019.e01802.
3. W. Awad and S. ELseuo, ``Machine learning methods for spam E-Mailclassification," Int. J. Comput. Sci. Inf. Technol., vol. 3, no. 1, pp. 173184,Feb. 2017, doi: 10.5121/ijcsit.2011.3112.
4. S. Mohammed, O. Mohammed, and J. Fiaidhi, ``Classifying unsolicitedbulk email (UBE) using Python machine learning techniques," Int.J. Hybrid Inf. Technol.Available:https://www.researchgate.net/publication/236970412_Classifying_Unsolicited_Bulk_Email_UBE_using_Python_Machine_Learning_Techniques
5. A. Wijaya and A. Bisri, ``Hybrid decision tree and logistic regressionclassifier for email spam detection," in Proc. 8th Int. Conf. Inf. Technol.Electr. Eng. (ICITEE), Oct. 2016, pp. 14, doi: 10.1109/ICITEED.2016.7863267.

6. K. Agarwal and T. Kumar, ``Email spam detection using integratedapproach of Naïve Bayes and particle swarm optimization," in Proc. 2nd Int. Conf. Intell.Comput. Control Syst. (ICICCS), Jun. 2018, pp.
7. A. I. Taloba and S. S. I. Ismail, ``An intelligent hybrid technique ofdecision tree and genetic algorithm for E-Mail spam detection," in Proc.9th Int. Conf. Intell. Comput. Inf. Syst. (ICICIS), Cairo, Egypt, Dec. 2019,pp. 99104, doi: 10.1109/ICICIS46948.2019.9014756.

WEBSITES:
1. https://www.w3schools.com/python/
2. https://www.javatpoint.com/python-tutorial
3. https://www.leanpython.org/
4. https://www.pythontutorial.net/