IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

An Effective Machine Learning-Based Spam Detection Method for IoT Devices

[1]Potbhare Nitin Balasaheb, [2]Prof. Sushil Venkatesh Kulkarni

[1]Potbhare Nitin Balasaheb, [2]Prof. Sushil Venkatesh Kulkarni

[1]Student, [2]Project Guide
[1]Computer Science & Engineering Technology,
[1]M.B.E. Society's College of Engineering, Ambajogai, India

Abstract:
Millions of devices with sensors and actuators connected via wired or wireless channels for data transmission make up the Internet of Things (IoT). By 2020, it is anticipated that over 25 billion devices will be connected, reflecting the IoT's tremendous growth over the last ten years. In the upcoming years, the amount of data released from these devices will multiply many-fold. In addition to producing more data overall, IoT devices also produce a lot of data in a variety of different modalities, with variable degrees of data quality determined by the speed of time and position dependency. In such a setting, machine learning algorithms can be crucial in assuring biotechnology-based security and authorisation as well as anomaly detection to enhance IoT systems. However, hackers frequently use learning algorithms to attack the flaws in IoT-based smart systems. In this study, we suggest employing machine learning to detect spam in order to secure IoT devices. Spam Detection in IoT utilising a Machine Learning framework is suggested to accomplish this goal. In this approach, a huge number of input feature sets are used to evaluate five machine learning models using a variety of criteria. Each model uses the enhanced input attributes to calculate a spam score. This rating shows how trustworthy Internet of Things (IoT) devices are based on several factors. The proposed technique is validated using the REFIT Smart Home dataset. In comparison to other current systems, the findings collected demonstrate the effectiveness of the proposed method.

Index Terms **–** PCA, MAE, MSE, RMSE.

I. INTRODUCTION

Regardless of their geographical locations, Internet of Things (IoT) facilitates convergence and implementations amongst real-world items. Implementing these network management and control mechanisms makes privacy and protection strategies crucial and difficult in this setting. To address security problems like invasions, spoofing attacks, DoS attacks, and jamming, and eavesdropping, spam, and malware, IoT applications must secure user data privacy. The size and kind of the business where IoT is implemented will determine the safety measures that are used. Users' actions force the security gateways to collaborate. In other words, we may say that the security measures for IoT devices are determined by their location, nature, and application.

The smart organization's IoT security cameras, for instance, can record many parameters for analysis and wise decision-making. The greatest amount of caution should be used with web-based devices because they make up the majority of IoT devices. IoT devices placed in an organisation are frequently utilised in the workplace to effectively implement security and privacy features. For instance, wearable technology that gathers and sends user health data to a connected smartphone should guard against data leaks to preserve privacy. According to market research, 25–30% of employees who are currently at work connect their personal IoT devices to the company network. Both consumers and attackers are drawn to the IoT because of its increasing nature. IoT devices pick a defensive approach and determine the essential parameters in the security protocols for a trade-off between security, privacy, and computation when ML emerges in various attack scenarios. It is difficult for an IoT system with limited resources to estimate the current network and timely attack status, thus this job is complex.

II. LITERATURE REVIEW

IoT systems, which include devices, services, and networks, are susceptible to network, physical, and application threats as well as privacy breaches.

- **Denial of service (DDoS) attacks:** To prevent IoT devices from accessing various services, the attackers can flood the target database with erroneous requests. Bots are a term used to describe these malicious requests made by an IoT device network. DDoS has the ability to deplete every resource offered by the service provider. It has the power to disable legitimate users and disable network resources.
- **RFID attacks:** These are the assaults launched on the IoT device's physical layer. The device's integrity is compromised as a result of this assault. Attackers make an effort to alter the data either at the node storage or during network transfer. Assaults on availability, attacks on authenticity, attacks on secrecy, and brute-forcing of cryptographic keys are frequent threats to the sensor node. Password protection, data encryption, and restricted access control are examples of countermeasures to assure the prevention of such assaults.
- **Internet attacks:** The Internet of Things gadget can remain connected to access a variety of resources. Spammers utilise spamming strategies when they wish to steal information from other systems or keep getting people to visit their target website. Ad fraud is a typical method employed for the same. For financial gain, it creates phoney clicks on a particular website. Cyber criminals are a group like this that practise online.
- **NFC attacks:** The major target of these assaults is fraud involving electronic payments. Unencrypted traffic, eavesdropping, and tag alteration are examples of potential assaults. Conditional privacy protection is the answer to this issue. As a result, the attacker is unable to generate the identical profile using the user's public key. The trusted service manager's random public keys serve as the foundation for this concept. Network security has been significantly improved using a variety of machine learning techniques, including supervised learning, unsupervised learning, and reinforcement learning.

Each machine learning method is discussed here along with its nature and function in attack detection.

- **Supervised machine learning techniques:** The models used for labelling the network for attack detection include support vector machines (SVMs), random forests, naive Bayes, K-nearest neighbour (K-NN), and neural networks (NNs). These models effectively identified DoS, DDoS, intrusion, and malware assaults on IoT devices.
- **Unsupervised machine learning techniques:** In the absence of labels, these strategies perform better than those of their counterparts. Clustering is how it functions. Multivariate correlation analysis is utilised in IoT devices to find DoS attacks.
- **Reinforcement machine learning techniques:** These models give an IoT system the ability to choose security protocols and crucial settings by testing them against various assaults. Both malware detection and authentication performance have benefited from the usage of Q-learning.

In order to conserve energy and increase the lifespan of the IoT system, machine learning approaches aid in the development of protocols for light-weight access control. For instance, the developed outer detection strategy uses K-NNs to overcome the problem of uncontrolled outer detection in WSNs. The review of the literature indicates how machine learning can be used to improve network security. As a result, in this research, multiple machine learning techniques are used to detect the presented problem of web spam.

III. PROPOSED SCHEME:

A. System model

Smart devices are entirely necessary for the digital age. These gadgets should only return information that is accurate and not spam. Because data is gathered from multiple domains, information retrieval from various IoT devices is a significant difficulty. IoT generates a vast number of heterogeneous, diverse data due to the involvement of several devices. This data can be referred to as IoT data. Real-time, multi-source, rich, and sparse are just a few of the characteristics of IoT data. If IoT data is saved, processed, and retrieved in an effective manner, TPR).

B. Proposed method:

This proposal focuses on online spam detection to prevent IoT devices from creating dangerous information.

For the purpose of detecting spam from IoT devices, many machine learning methods have been taken into consideration. The goal is to fix the problems with IoT devices installed in homes.

However, the suggested approach takes into account every aspect of data engineering before validating it using machine learning models. The measures taken to reach the goal are outlined one at a time as follows:

(1) **Feature Engineering**: The relevant instances and their attributes are required for the machine learning algorithms to function accurately. We are all aware that the instances represent actual data values from deployed smart devices in the real world. The basis of the feature engineering process is feature extraction and feature selection.

- **Feature reduction:** The dimension of the data is reduced using this methodology. In other terms, the process of reducing feature complexity is known as feature reduction. This method lessens problems with over-fitting, high memory needs, and computational power. There are numerous methods for feature extraction. The most well-liked of them is principal component analysis (PCA). However, this suggestion uses PCA combined with the IoT characteristics listed below.

  - **Analysis period:** The dataset utilised in the experiments includes data gathered over a period of 18 months. We took into account data from one month in order to get better outcomes and accuracy. The month with the greatest differences has been taken into consideration because the climate is a crucial factor in how well an IoT device functions.

  -Appliances that are web-based are those that require a constant internet connection to function. The following appliances are part of the data collection: TV, set top box, DVD player/recorder, HiFi, electric heater, fridge, dishwasher, toaster, coffee maker, kettle, freezer, washing machine, tumble dryer, electric heater, DAB radio, desktop computer, PC monitor, printer, router, electric heater, shredder, freezer, CD player, TV, video player, set top box, hub (network).

- **Feature selection:** It entails calculating the most significant subset of features. It operates by calculating the value of each feature. In this proposal, the feature selection method is an entropy-based filter.

  - Filter based on entropy: To determine the weights of discrete qualities, this technique analyses the correlation between discrete and continuous attributes. This entropy-based filter uses three different functions: information.gain, gain.ratio, and symmetrical.uncertainty. These functions have the following syntax:

  information.gain (formula, data, unit)

  gain.ratio (formula, data, unit)

  symmetrical.uncertainty (formula, data, unit)

  The function definition's arguments are described.'

  1. **Formula:** This is a description of how the algorithm functions.
  2. **Data:** The set of training data with the specified properties is what will be used to make the selection.
  3. **Unit:** It is the metric used in entropy calculations. It accepts the value "log" by default.

C. Machine learning models: By employing a machine learning methodology to identify the spam parameters, the suggested method is proven to work.

(1) ***Bayesian Generalized Linear Model (BGLM):*** For exponential family forms, it is a consistent, asymptotically effective, and asymptotically normal log likelihood uni-modal. The focus of Bayesian approaches really lies on these fundamental components.

• Previous data is integrated first. Typically, previous knowledge is quantified in the form of a distribution, which indicates the probability distribution for a coefficient.

• Next, a function of likelihood is linked with the prior.

The outcomes are represented by the probability function.

• Third, a distribution of coefficient values is created as a result of the prior and probability function combined.

• Fourth, simulations from the posterior distribution are used to create an empirical distribution of likely values for the population parameter.

• Fifth, basic statistics are employed to summarise the statistical distribution of simulations from the posterior.

(2) **Boosted linear model:** Multiple decision trees are generated for the data pieces, and the decision tree models are created by categorising the data series into various data classes. Consequently, each of the data groupings is modelled as a linear function.

(3) **EXtreme Gradient Boosting (xgboost):** It is a scalable and effective gradient boosting technique. An efficient linear model solver and a tree learning technique are also included in the package. It offers a number of objective operations including ranking, grouping, and regression. It functions with vectors of numbers. Compared to current gradient boosting methods, it is five times faster. To determine the optimum tree model, the gradient boosting method makes advantage of more precise approximations. It employs a variety of cunning techniques that set it apart from structured data in general. Each training cycle builds up the weak learner, and its predictions are matched with the appropriate results. Our model's error rate is the discrepancy between prediction and reality. These inaccuracies can be used to determine the gradient. The gradient defines the steepness of the error function, but it does nothing unique other than being the partial derivative of the loss function. The gradient can be used to determine how to modify the system's settings such that the error in the following cycle of learning can be "downgraded" as much as possible.

This model was created using the following formula.

xgb ← xgboost(data , label, eta, max depth, nround, subsample, colsample bytree, seed, eval metric, objective, num class, nthread).

This approach primarily uses three different kinds of parameters: general parameters (booster, num class, etc.), booster parameters (max depth, gamma, etc.), and learning task parameters (base score, objective...).

(4) **Generalized Linear Model with Stepwise Feature Selection:** A number of explanatory (predictor) variables can be used to interpret a dependent variable using generalised linear models (GLMs), which offer a dynamic framework for doing so. The explanatory variables can either be empirical (co-variate) or categorical, and the parameter dependant can either be continuous or discrete (factors). The model was fitted using a stepwise feature selection process. It is necessary to use this procedure repeatedly until substantial results are obtained for each of the equation's effects. When using R, the support glmulti function is used to specify the equation.

IV. RESULTS:

The suggested method finds the spam parameters that are affecting IoT devices. The suggested approach is validated using the IoT dataset, as stated in the next Section, to achieve the best results.

A. Data Gathering:

We gathered the smart house dataset through the REFIT project. Twenty homes in total were used and given the go-ahead to implement smart home technologies. The team of researchers conducted the entire poll. Depending on factors like floor plans, Internet accessibility, and climatic fluctuations, the trials vary from room to room. Utilizing several sensors, the internal environmental conditions were recorded. Every residence had sensors that could monitor more than 100,000 data points. The survey kept going for almost 18 months.

B. Experimental setup: We use the given data set traces from the source to carry out the studies. The trials were then run using RStudio. The necessary software is, Windows 7/8/10, MacOS 10.12+, Ubuntu 14/16/18, or Debian 8/10 are the supported operating systems. **Language**: Python, **Front End**: Anaconda Navigator – **Spyder IDE.**

The findings are listed below.

C. Impact of data preprocessing on SDI-UML: The preprocessing entails choosing the appliances under consideration for spam detection settings. The major goal is to identify the numerous elements that cause spam. The feature reduction is completed first. Principal Component Analysis (PCA), which decreases the dimensions of data, is a technique for feature reduction. It generates a set of Principal Components (PC) for every row and every column. We have 15 features in the IoT dataset we used for our project, hence 15 PCs are generated. The pca() decreases the variance among the features through the manner it operates.

D. Impact of machine learning models on SDI-UML: Five distinct machine learning models are trained on the dataset using the aforementioned features. Each model generates a spamicity score for each appliance that quantifies the likelihood that the appliance may be impacted by spam. To determine the accuracy, precision, and recall, an evaluation is conducted.

```
------------------------------------------------------
================== Data Selection ====================
------------------------------------------------------

         time  use [kW]  gen [kW]  ...  precipIntensity  dewPoint  precipProbability
0  1451624400  0.932833  0.003483  ...              0.0      24.4                0.0
1  1451624401  0.934333  0.003467  ...              0.0      24.4                0.0
2  1451624402  0.931817  0.003467  ...              0.0      24.4                0.0
3  1451624403  1.022050  0.003483  ...              0.0      24.4                0.0
4  1451624404  1.139400  0.003467  ...              0.0      24.4                0.0
5  1451624405  1.391867  0.003433  ...              0.0      24.4                0.0
6  1451624406  1.366217  0.003450  ...              0.0      24.4                0.0
7  1451624407  1.431900  0.003417  ...              0.0      24.4                0.0
8  1451624408  1.627300  0.003417  ...              0.0      24.4                0.0
9  1451624409  1.735383  0.003417  ...              0.0      24.4                0.0

[10 rows x 32 columns]
```

Fig. 1. Data Selection

```
------------------------------------------------------
============ Before Checking Missing Values ==========
------------------------------------------------------

time                  0
use [kW]              1
gen [kW]              1
House overall [kW]    1
Dishwasher [kW]       1
Furnace 1 [kW]        1
Furnace 2 [kW]        1
Home office [kW]      1
Fridge [kW]           1
Wine cellar [kW]      1
Garage door [kW]      1
Kitchen 12 [kW]       1
Kitchen 14 [kW]       1
Kitchen 38 [kW]       1
Barn [kW]             1
Well [kW]             1
Microwave [kW]        1
Living room [kW]      1
Solar [kW]            1
temperature           1
icon                  1
humidity              1
visibility            1
summary               1
```

```
------------------------------------------------------
============ Before Checking Missing Values ==========
------------------------------------------------------

time                  0
use [kW]              1
gen [kW]              1
House overall [kW]    1
Dishwasher [kW]       1
Furnace 1 [kW]        1
Furnace 2 [kW]        1
Home office [kW]      1
Fridge [kW]           1
Wine cellar [kW]      1
Garage door [kW]      1
Kitchen 12 [kW]       1
Kitchen 14 [kW]       1
Kitchen 38 [kW]       1
Barn [kW]             1
Well [kW]             1
Microwave [kW]        1
Living room [kW]      1
Solar [kW]            1
temperature           1
icon                  1
humidity              1
visibility            1
summary               1
```

Fig. 2. Before Checking Missing values

```
------------------------------------------------
=============== Before label encoding ===========
------------------------------------------------

0   clear-night
1   clear-night
2   clear-night
3   clear-night
4   clear-night
5   clear-night
6   clear-night
7   clear-night
8   clear-night
9   clear-night
Name: icon, dtype: object
```

```
------------------------------------------------
=============== After label encoding ===========
------------------------------------------------

0   2
1   2
2   2
3   2
4   2
5   2
6   2
7   2
8   2
9   2
Name: icon, dtype: int32
```

Fig. 3. Before local encoding

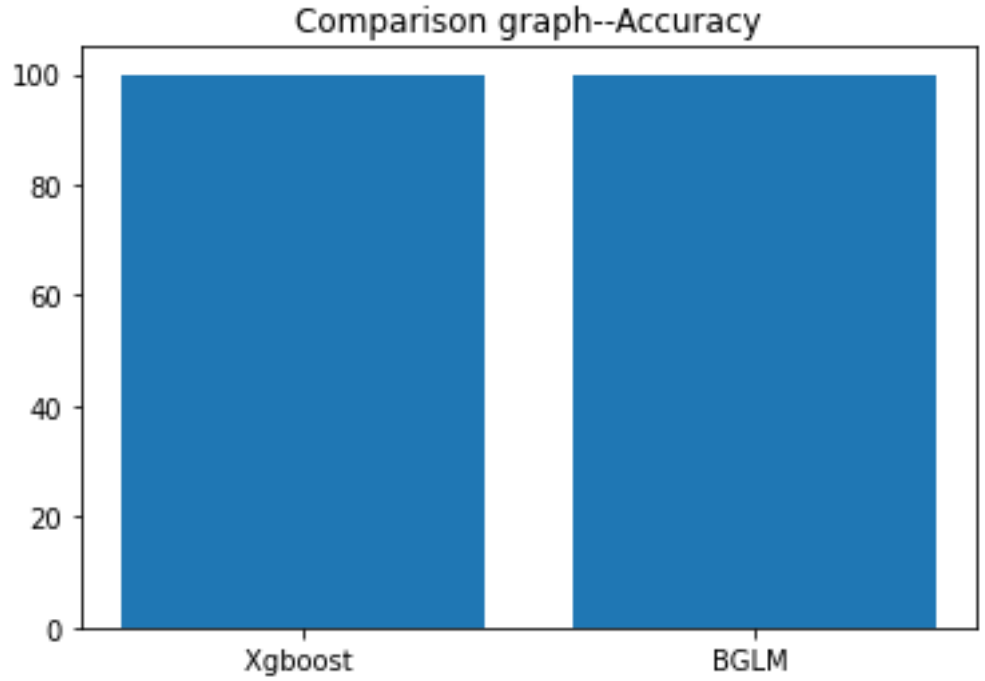

Fig. 4. Comparison graph - Accuracy

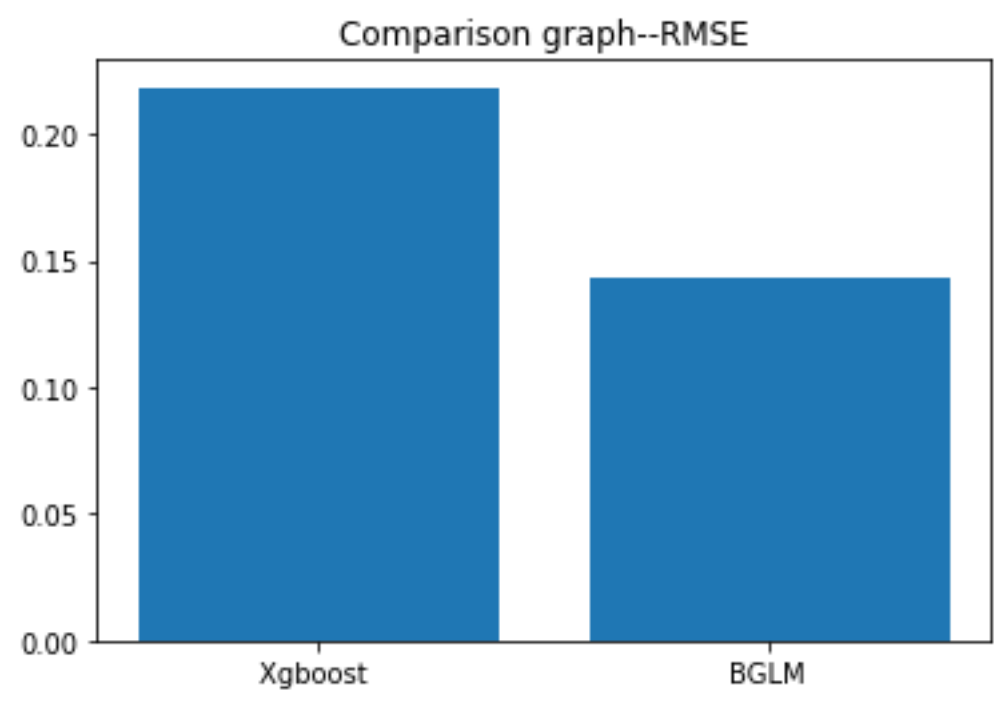

Fig. 5. Comparison graph - RMSE

V. CONCLUSION

The suggested system uses machine learning models to find spam parameters in IoT devices. The feature engineering technique is used to pre-process the IoT dataset that will be used in the trials. Each IoT gadget is given a spam score by testing the framework with machine learning models. This clarifies the prerequisites needed for IoT devices in a smart home to operate effectively. In the future, we intend to take into account the environmental characteristics of IoT devices to increase their reliability and security.

ACKNOWLEDGMENT

I pay thanks to our project guide Prof. Sushil Venkatesh Kulkarni for assistance and guidance especially related to technicalities and also who encouraged and motivated us.

REFERENCES

[1] An Efficient Spam Detection Technique for IoT Devices using Machine Learning, Dr. Aaisha Makkar, Student Member, IEEE, Dr. Sahil (GE) Garg, Senior Member, IEEE Dr. Neeraj Kumar, Senior Member, IEEE Prof. M. Shamim Hossain § , Senior Member, IEEE Prof. Ahmed Ghoneim, Senior Member, IEEE Dr. Mubarak Alrashoud k , Senior Member, IEEE ∗Computer Science and Engineering Department, Chandigarh University (Punjab), India (aaisha.e8847@cumail.in).

[2] Z.-K. Zhang, M. C. Y. Cho, C.-W. Wang, C.-W. Hsu, C.-K. Chen, and S. Shieh, "Iot security: ongoing challenges and research opportunities," in 2014 IEEE 7th international conference on service-oriented computing and applications. IEEE, 2014, pp. 230–234.

[3] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in 2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops). IEEE, 2017, pp. 618–623.

[4] E. Bertino and N. Islam, "Botnets and internet of things security," Computer, no. 2, pp. 76–79, 2017.

[5] C. Zhang and R. Green, "Communication security in internet of thing: preventive measure and avoid ddos attack over iot network," in Proceedings of the 18th Symposium on Communications & Networking. Society for Computer Simulation International, 2015, pp. 8–15.

[6] W. Kim, O.-R. Jeong, C. Kim, and J. So, "The dark side of the internet: Attacks, costs and responses," Information systems, vol. 36, no. 3, pp. 675–705, 2011.

[7] H. Eun, H. Lee, and H. Oh, "Conditional privacy preserving security protocol for nfc applications," IEEE Transactions on Consumer Electronics, vol. 59, no. 1, pp. 153–160, 2013.

[8] R. V. Kulkarni and G. K. Venayagamoorthy, "Neural network based secure media access control protocol for wireless sensor networks," in 2009 International Joint Conference on Neural Networks. IEEE, 2009, pp. 1680–1687.

[9] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan, "Machine learning in wireless sensor networks: Algorithms, strategies, and applications," IEEE Communications Surveys & Tutorials, vol. 16, no. 4, pp. 1996– 2018, 2014.

[10] Z. Tan, A. Jamdagni, X. He, P. Nanda, and R. P. Liu, "A system for denial-of-service attack detection based on multivariate