Using the K-Nearest Neighbour and Moth Blade Optimization Algorithm to Identify Malicious Sessions in IoT Networks

KAMEPALLI UMA[1], T HARI BABU[2]

Assistant professor[1,2]

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
P.B.R. VISVODAYA INSTITUTE OF TECHNOLOGY & SCIENCE
S.P.S.R NELLORE DIST, A.P, INDIA, KAVALI-524201.

To Cite this Article

KAMEPALLI UMA[1], | T HARI BABU| "Using the K-Nearest Neighbour and Moth Blade Optimization Algorithm to Identify Malicious Sessions in IoT Networks" *Journal of Science and Technology, Vol. 07, Issue 10, - oct 2022, pp1-9*

Article Info

Received: 04-11-2022 Revised: 05-11-2022 Accepted 15-12-2022 Published: 27-12-2022

Abstract-

There are many ways in which the convenience of an IOT network might improve people's daily lives. Due to the increasing number of potential targets, the security of IoT devices is a pressing issue of the present. In this study, we offer a method for detecting intrusions into IoT networks, which classifies sessions into either attack or regular categories. Work for slection of characteristics for determining the class representative sessions employed a moth flame optimization genetic method. K-Nearest Neighbor was used to determine which class meeting it was. The experimental results, which were obtained using a real dataset, demonstrate that the suggested model, Moth Flame based IOT Network Security (MFIOTNS), is able to optimise different values of the evaluation parameters to provide greater gains in productivity.

Keywords: KNN, Clustering, GA, and Intrusion Detection.

INTRODUCTION

As the use of computers and other forms of electronic communication becomes more commonplace, so do worries about infringements on personal privacy. New and recent attempts to enter computer networks and systems are a direct result of the explosion in the number of Internet-based applications and the emergence of cutting-edge technology like the Internet of Things (IoT). The Internet of Things (IoT) refers to a network of devices that may communicate with one another automatically, without any human involvement. The Internet of Things (IoT) allows a wide variety of sensor-equipped devices (including coffee makers, lights, bicycles, and many more) to communicate with one another and with the wider world through the Internet in fields as diverse as medicine, agriculture, transportation, and more [1]. Apps built for the Internet of Things are revolutionising our daily lives and the way we do business by helping us save both precious time and money. There are no limits to the benefits, and it provides a wealth of new possibilities for sharing information, fostering creativity, and advancing progress. Since the Internet serves as the backbone and nerve centre of the IoT, any security risk that exists on the Internet also exists inside the IoT. Nodes in an IoT network have restricted capabilities, less resources, and no user-managed settings or preferences.

With the proliferation of IoT devices and their incorporation into everyday life, security concerns have emerged as a major challenge, prompting the demand for network-based security solutions. While the state-of-the-art systems are excellent at spotting certain types of assaults, others remain difficult to see. There is no question that there is room for more innovative approaches to enhance network security, since the number of network assaults grows in tandem with the tremendous rise in the volume of information contained in networks [2]. In this regard, Machine Learning (ML) may be seen as one of the most efficient computational models for providing embedded intelligence in the IoT environment. Many network security activities, including traffic analysis, intrusion detection, and botnet identification, have benefited from the use of machine learning [3, 4, 5, 6, 7]. A crucial component of any Internet of Things (IoT) solution is machine learning, which can be defined as the capacity of a smart device to adapt to new situations and automate previously manual tasks based on the device's acquired knowledge. ML may infer useful information from data supplied by devices or people, and ML algorithms are utilised in tasks like regression and classification. For that matter, ML may also be utilised to safeguard an IoT network. There is a growing interest in applying ML to challenges of attack detection, and ML is finding a growing number of uses in the cybersecurity industry. There is a dearth of literature on effective detection approaches appropriate for IoT contexts, despite the widespread usage of ML techniques to uncover the best ways to detect assaults. Signature-based (often also termed misuse-based) and anomaly-based cyber-analysis are the two primary ways in which machine learning may be applied to the attack detection job. Methods that rely on "signatures," or unique patterns of communication, may identify known assaults. The capacity to efficiently identify all known threats without producing an excessive amount of false alarms is a major benefit of this kind of detection technology.

WORK IN RELATION

An intrusion detection approach using a genetic algorithm and a deep belief network is proposed in [13]. The NSL-KDD dataset is used for the detection of four distinct attacks: DoS, R2L, Probe, and U2R. In contrast to our work, this study does not use blockchain in their solution as an integrated method for monitoring and safeguarding IIoT networks, and it employs an outdated dataset that is not easily adaptable to contemporary IoT networks. In [14], we propose using a statistical flow characteristics-based intrusion detection method to guard IoT application network traffic. In this study, the authors use three different machine learning methods—a Decision Tree, a Naive Bayes classifier, and an Artificial Neural Network—to identify fraudulent traffic events (ANN). Although they utilise the UNSWNB15 dataset, which we also use, their solution does not use blockchain as an integrated method for monitoring and safeguarding IIoT networks. In [15], the authors offer a machine learning security architecture for Internet of Things (IoT) devices. They created their own dataset using data from the NSL KDD dataset and tested it in an actual smart building environment. As we discussed in the aforementioned works, an outdated dataset may not function for today's IoT systems. Their method for identifying DDoS, Probe, U2R, and R2L assaults is based on a single class of support vector machines (SVMs). Their method of monitoring IIoT networks is not blockchain-based, however. To identify distributed denial-of-service (DDoS) assaults, the developers of [16] created a deep-learning system. Random Forests, Multilayer Perceptron, and Convolutional Neural Network are the three methods used to identify Denial of Service assaults. While they use the same information that we do, their solution does not make use of blockchain technology and instead aims to identify a single kind of attack (DoS).

METHODOLOGY

In this part, we present a synopsis of the Moth Flame-based IoT Network Security Framework Proposal (MFIOTNS). The proposed model's processing, dimensionality reduction, and training blocks are shown in Fig. 1. This part had an explanation of each block organised under respective topics.

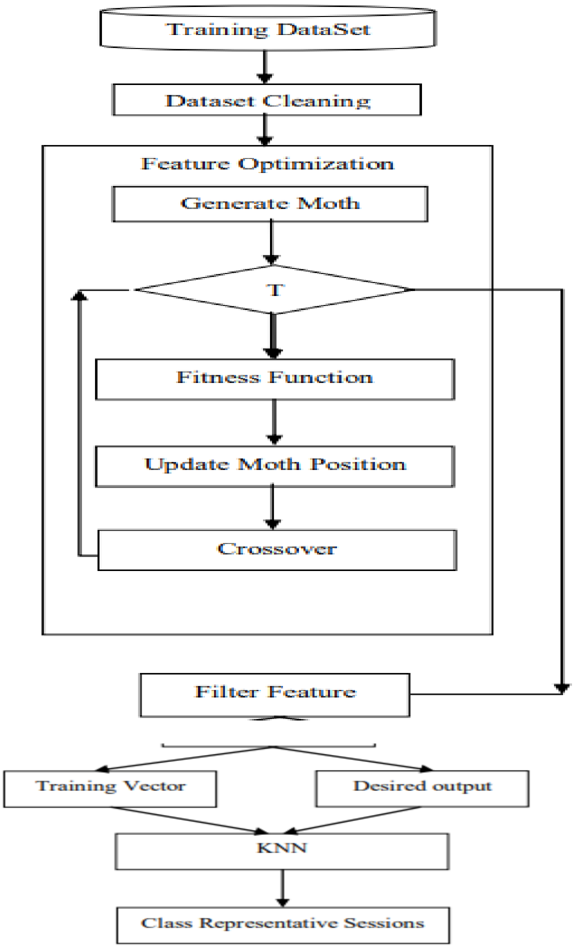


Fig. 1 Block diagram of MFOCMSD network intrusion detection.

$$CD \leftarrow Dataset\_Cleaning(RD) \text{ -----------Eq. 1}$$

In the first equation, RD represents the unprocessed data and CD represents the refined data. Dataset after processing was laid out as a matrix, with rows representing sessions and columns representing feature sets for those sessions.

Enhancing Specific Functions

The Moth Flame Optimization Algorithm was applied to the input CD matrix, which decreased the training vector values and improved the accuracy of the learning process.

The authors of this work developed a method called the Moth Flame Optimization Algorithm, in which each chromosome is represented by a moth. The goal of this programme was to locate a Moth Flame that was part of the route to the moon. The moth flames that populate this painting are its chromosomes.

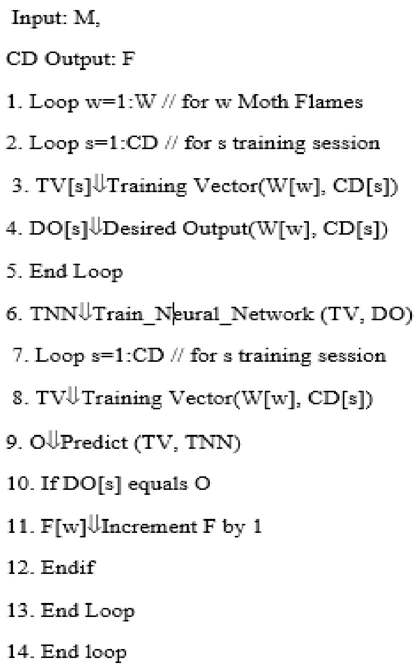
Create Flaming Moths

If you think of moths as a collection of chromosomes, then each flame represents a different viable solution to an optimal feature set. This means that the size of a Moth Flame is a vector with n elements, where n is the total number of columns in a CD. Moth Flame vector has two possible values for each digit. A value of one indicates that the characteristic is considered during training, whereas a value of zero indicates that it is not chosen during population construction. For this reason, if a population of Moth Flames is generated, M, the Moth Flame population matrix, will have pxn dimensions. The Gaussian random-value generator function is used to choose f features from the vector at random.

M← Generate_Moth Flame(p, n, f)----------Eq. 2

Fitness Function

Each Moth Flame were rank as per distance. So evaluation of distance done by fitness value. Moth Flame feature vector pass training vector to the KNN (K-Nearest Neighbour) for cluster representative finding and measure the detection accuracy of the work [11]. This detection accuracy value is distance parameter in the work.



```
Input: M,
CD Output: F
1. Loop w=1:W // for w Moth Flames
2. Loop s=1:CD // for s training session
3. TV[s]⇓Training Vector(W[w], CD[s])
4. DO[s]⇓Desired Output(W[w], CD[s])
5. End Loop
6. TNN⇓Train_Neural_Network (TV, DO)
7. Loop s=1:CD // for s training session
8. TV⇓Training Vector(W[w], CD[s])
9. O⇓Predict (TV, TNN)
10. If DO[s] equals O
11. F[w]⇓Increment F by 1
12. Endif
13. End Loop
14. End loop
```

In above algorithm TV is training vector, DO is desired output.

Position of the Moth Flame, please update

Assuming that a fitness function has been used to determine a F value, the best Moth Flame may be selected from the pool of accessible chromosomes by sorting the candidates in descending order according to their F values.

Crossover

The success of a genetic method relies on chromosomal changes; hence the values of Moth Flames' random positions were adjusted when the parameter X was adjusted. Moth Flame standards were not upheld throughout this procedure. Here, the finest local Moth Flame feature set was used to arbitrarily flip X locations on each Moth Flame from zero to one, or one to zero. These Moth Flames were put through further tests, including a comparison of their fitness levels to those of their parents. If the maximum number of iterations has been reached, go to the filter feature block; otherwise, calculate the fitness of each Moth Flame.

Functionality for Applying Filters

Upon completion of the iteration, the best Moth Flame will be selected from the most recent population. One-valued features in a chromosome are considered to be the "chosen" features for use in creating the training vector, whereas zero-valued features are not. In this part we also constructed the output matrix that we wanted.

K-means clustering based representativeness

The KNN model was then used to identify a representative member of each cluster using the feature set acquired from the aforementioned technique. A session's class may be determined from the distance vector to the representative feature set, hence finding such a representative is useful.

A TRY IT OUT AND THE END RESULT

The experimental framework was constructed in MATLAB, where both the MFOCMSD and comparison models were written. i3 6th generation CPU, 4 GB of RAM; this is an experimental system. Dataset used for IO was obtained from [15]. The MFIOTNS model was compared to a model for detecting malicious sessions in the cloud that was developed in [16].

Metrics for Assessment

In order to evaluate our work, we use the metrics of Precision, Recall, and F-score. True positives, false positives, and false negatives (TP, TN, FP, and FN) all play a role in determining these values (False Negative).

RESULTS

Table 1. Precision value-based comparison of IOT network intrusion detection models.

Dataset Size	Previsous Work	MFIOTNS
5000	0.9359	0.987
10000	0.9322	0.9814
15000	0.9312	0.9812
20000	0.9322	0.9806
25000	0.9323	0.9793

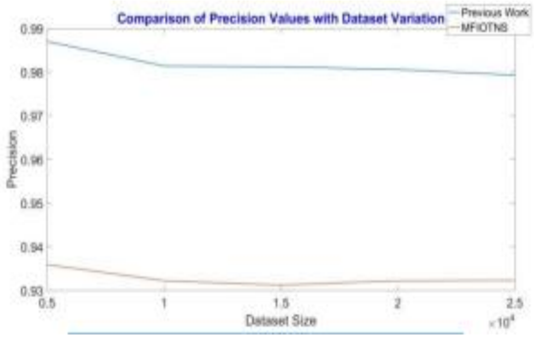


Fig. 2 Precision value-based comparison.

IOT network intrusion detection models were compared on different dataset size and result shows that prosed model has improved the precision value by 5.004% as compared to previous model proposed in [16]. It was found that that proposed model has increases the precision value by use of moth flame feature optimization technique, as less feature has improved the clustering of KNN model.

Table 2. Recall value-based comparison of IOT network intrusion detection models

Dataset Size	Previsous Work	MFIOTNS
5000	0.8623	0.9862
10000	0.8568	0.9838
15000	0.8582	0.9825
20000	0.8586	0.9816
25000	0.8606	0.9816

Recall value parameters were compared in table 2. It was obtained that proposed model has improved the IOT intrusion detection recall parameter by % as compared to values obtained from the previous model in [16]. KNN based learning of selected feature has increases the detection recall.

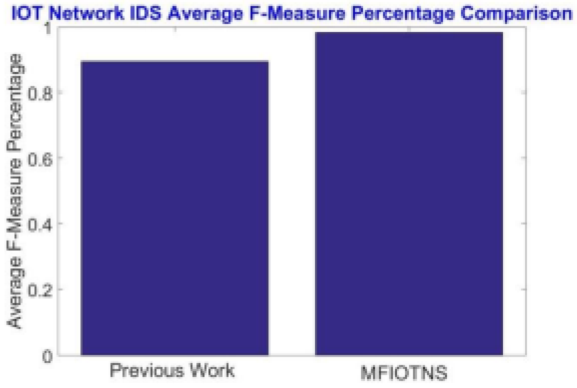
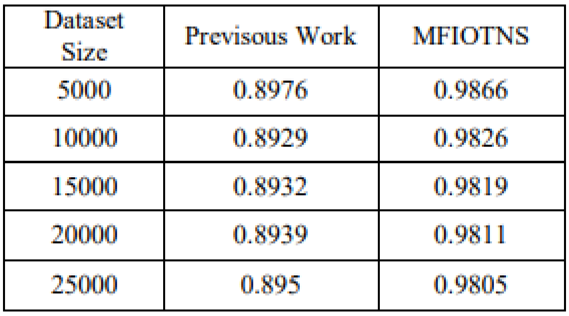


Fig. 3 F-measure value-based comparison.

Table 3. F-Measure value-based comparison of IOT network intrusion detection models.

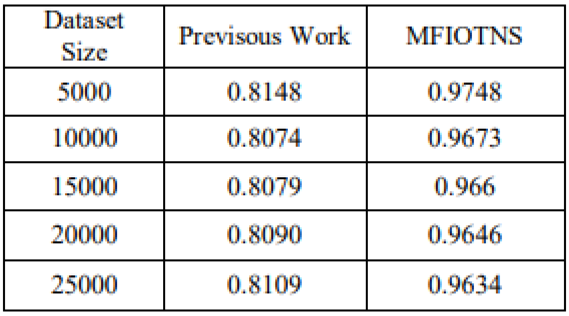


Dataset Size	Previsous Work	MFIOTNS
5000	0.8976	0.9866
10000	0.8929	0.9826
15000	0.8932	0.9819
20000	0.8939	0.9811
25000	0.895	0.9805

Inverse average of precision and recall value is f-measure parameter.

Table 3 shows that use of moth flame optimization genetic algorithm for feature selection has enhanced the f-measure values of IOT network intrusion detection.

Table 4. Accuracy value-based comparison of IOT network intrusion detection models.



Dataset Size	Previsous Work	MFIOTNS
5000	0.8148	0.9748
10000	0.8074	0.9673
15000	0.8079	0.966
20000	0.8090	0.9646
25000	0.8109	0.9634

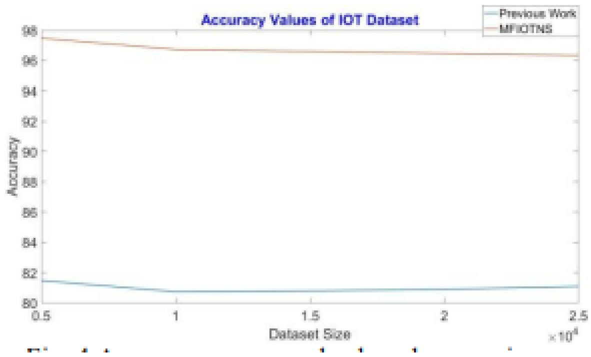


Fig. 4 Average accuracy value-based comparison.

The accuracy value of the prosed model has increased by 6.25 percentage points in comparison to the prior model published in [16], according to a comparison of models for intrusion detection in IoT networks conducted on datasets of varying sizes. Utilizing moth flame feature optimization, it was discovered that the suggested model improves the clustering of the KNN model with less features, leading to higher accuracy.

CONCLUSION

Small businesses, hotels, organisations, etc., benefit greatly from network connections. However, due to insufficient security mechanisms, it is susceptible to a wide variety of assaults. In this research, we provide a model for such a network's intrusion detection capabilities. Datasets with feature sets and vole sets are used as input. The moth flame optimization method sorts these characteristics into groups of those to be kept and those to be discarded. For both intrusion and non-intrusion class detection, the KNN method was applied to the selected features to determine the feature values that serve as the cluster centres. The results of an experiment conducted on the IOT dataset demonstrate that the suggested model is superior to the state-of-the-art in terms of intrusion detection accuracy by a significant margin. Scholars in the future may improve the work's detection accuracy by employing a different training model.

REFERENCES

[1] .J. *Deogirikar and A. Vidhate, "Security attacks in iot: A survey," International Conference on I-SMAC (ISMAC), pp. 32–37, 2017.*

[2]. T. Bodstrom and T. H ̈ am ̈ al ̈ ainen, "State of the art literature review ̈ on network anomaly detection with deep learning," Internet of Things, Smart Spaces, and Next Generation Networks and Systems, pp. 64–76, 2018.

[3] .I. Arnaldo, A. Cuesta-Infante, A. Arun, M. Lam, C. Bassias, and K. Veeramachaneni, "Learning representations for log data in cybersecurity," International Conference on Cyber Security Cryptography and Machine Learning, pp. 250–268, 2017.

[4]. M. Du, F. Li, G. Zheng, and V. Srikumar, "Deeplog: Anomaly detection and diagnosis from system logs through deep learning," Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1285–1298, 2017.

[5] . B. J. Radford, B. D. Richardson, and S. E. Davis, "Sequence aggregation rules for anomaly detection in computer network traffic," arXiv preprint arXiv:1805.03735, 2018.

[6]. I. Lambert and M. Glenn, "Security analytics: Using deep learning to detect cyber attacks," 2017.

[7]. M. Stevanovic and J. M. Pedersen, "Detecting bots using multi-level traffic analysis." IJCSA, vol. 1, no. 1, pp. 182–209, 2016.

[8]. Zhang, Y.; Li, P.; Wang, X. Intrusion Detection for IoT Based on Improved Genetic Algorithm and Deep Belief Network. IEEE Access 2019, 7, 31711–31722.

9. Moustafa, N.; Turnbull, B.; Choo, K.R. An Ensemble Intrusion Detection Technique based on proposed Statistical Flow Features for Protecting Network Traffic of Internet of Things. IEEE Internet Things J. 2018, 6, 4815–4830.

[10]. Bagaa, M.; Taleb, T.; Bernal, J.; Skarmeta, A. A machine learning Security Framework for Iot Systems. IEEE Access 2020, 8, 114066–114077.

[11]. Susilo, B.; Sari, R. Intrusion Detection in IoT Networks Using Deep Learning Algorithm. Information 2020, 11, 279.

[12]. Liu, J.; Kantarci, B.; Adams, C. Machine LearningDriven Intrusion Detection for Contiki-NG-Based IoT Networks Exposed to NSL-KDD Dataset. In Proceedings of the ACM Workshop on Wireless Security and Machine Learning, Linz, Austria, 13 July 2020.

[13].Pokhrel, S.; Abbas, R.; Aryal, B. IoT Security: Botnet detection in IoT using Machine learning. arXiv 2021, arXiv:2104.02231.

[14] .21. Shafiq, M.; Tian, Z.; Sun, Y.; Du, X.; Guizani, M. Selection of effective machine learning algorithm and Bot-IoT attacks traffic identification for internet of things in smart city. Future Gener. Comput. Syst. 2020, 107, 433–442.

[15].Ullah I., Mahmoud Q.H. (2020) A Scheme for Generating a Dataset for Anomalous Activity Detection in IoT Networks. In: Goutte C., Zhu X. (eds) Advances in Artificial Intelligence. Canadian AI 2020.

[16] .A. Fatani, M. Abd Elaziz, A. Dahou, M. A. A. AlQaness and S. Lu, "IoT Intrusion Detection System Using Deep Learning and Enhanced Transient Search Optimization," in IEEE Access, vol. 9