International Journal on Advances in Engineering Technology and Science
Volume: 5, Issue: 2, 2024, e-ISSN: 2455-3131
DOI: 10.5281/zenodo.14408238

Deep Learning and Elephant Herd Based IOT Network Intrusion Alarming System

Diwakar Kumar Chaudhary[1], Pritaj Yadav[1], Kanchan Jha[2]

[1]Computer Science and Engineering, Ravindranath Tagore University, Bhopal, India
[2]J.S.C.S H/S Parsauni Murliyachak Madhubani, India

***Abstract*—The Internet of Things (IoT) have various applications in different sectors of with each passing day. The rapid development of IoT and its increasing demand in different fields of life create a serious problem of security for the IoT environment, which needs serious consideration to protect the IoT-enabled systems from external networks and cyber-attacks.IDS solutions are now more essential than ever because of the rising use of Internet of Things (IoT) devices. This paper proposes a deployment-ready network IDS for IoT that overcomes the shortcomings of the existing IDS solutions. This paper has proposed a model for IOT network intrusion detection using optimized features. In order to optimize data Elephant Herd optimization algorithm was developed. Filtered features were further process by convolutional and maxpooing operator to train the deep neural network. Experiment was done on real dataset and result shows that proposed DEIIDS (Deep Learning & Elephant based IOT Intrusion Detection System) model successfully detect intrusion and its classes.**

Keywords- Deep Learning, Intrusion Detection, Feature Optimization, Genetic Algorithm, Soft Computing.

I. INTRODUCTION

The Internet of Things, commonly known as IoT, constitutes an interconnected system relying on approved protocols for information exchange. Ongoing advancements in this field contribute significantly to the evolution of intelligent entities such as smart cities, devices, homes, transportation, healthcare, agriculture, grids, military applications, and more [1]. IoT serves as the foundation for interfacing with real-world applications over the Internet within its domain [2]. Furthermore, the Industrial Internet of Things gains momentum through the rapid integration of sensors and devices [2]. Consequently, the deployment of machine learning, deep learning, and artificial intelligence becomes imperative to furnish intelligent solutions for industrial edge computing [3].

The generation of traffic data in IoT networks originates from sensors, which then transmit data through wireless or wired communication channels. Consequently, the IoT communication channel must efficiently handle massive traffic volumes from a plethora of devices and sensors, ensuring nearly zero packet drops during transmission and bolstered protection at external edges [1,2]. Given the resource limitations of IoT devices and the unattended operational environment, security and protection mechanisms for these systems/networks need to be both efficient and effective.

In addition to grappling with information storage and management challenges, security and privacy concerns—encompassing confidentiality, integrity, and availability—stand out as crucial requirements during the early architectural design of a smart system. IoT networks encounter diverse security threats that can jeopardize the core functionalities of data storage and communication, putting user data privacy at risk. Attacks such as Denial of Service (DoS) can render entire networks or specific nodes unavailable, disrupting the flow of information. Other attacks like scanning, enumeration, and reconnaissance enable privacy compromise by revealing sensitive data, undermining various security facets associated with IoT networks.

Within wireless sensor networks, security issues present a major challenge [1]. An Intrusion Detection System (IDS) serves as software designed to monitor a computer network or system for signs of unauthorized access or violations of security policies. Common and growing practices include notifying an administrator or centrally logging security infractions and events through a Security Information and Event Management (SIEM) system.

The subsequent sections of the paper are organized into four parts. The next section provides a brief overview of intrusion detection models proposed by other researchers. Following that, the third section outlines the proposed model. The fourth section presents experimental results of the proposed model across different evaluation parameters. Finally, the paper concludes with various findings and outlines potential avenues for future work.

II. RELATED WORK

The authors [7] propose a tool based on virtual machine introspection (VMIA) to identify malicious mining within the same operating system. This approach addresses the challenge of mining software evading detection systems and the current inadequacy of protection against widespread binary mining software using the virtual machine introspection (VMI) technique for malware detection and defense.

In [8], Kumar R. et al. introduce a distributed Intrusion Detection System (IDS) designed to identify Distributed Denial of Service (DDoS) attacks in IoT networks. Leveraging fog computing and blockchain network architecture, this IDS deploys intrusion detection components on fog computing resources. Each component is responsible for detecting intrusions within a group of

IoT devices. The model facilitates the flow of transactional information of fog computing resources through the blockchain network. Random forest and XGBoost models are also incorporated in the learning components for effective intrusion detection.

Le K. H. et al. [9] present an IDS for IoT that employs deep learning, specifically a Convolutional Neural Network (CNN). To overcome limited training data, they integrate a Generative Adversarial Network (GAN) into the study. This neural network generates artificial training samples to comprehensively simulate attack behaviors based on real training samples.

In [10], Nasr et al. investigate sixteen Electric Vehicle Charging Station Management Systems (EVCSMSs) from reputable vendors deployed mainly in Europe and the United States. The study reveals critical zero-day vulnerabilities in web, mobile, and firmware aspects. Exploiting these vulnerabilities, the researchers compromise the EVCS, demonstrating practical implications against both the EVCS and its users. This study highlights the lack of academic attention to EVCS firmware and EVCSMS security in comparison to other components of the EV ecosystem. It emphasizes the vulnerability of live systems used for charging electric vehicles, suggesting mitigations to reduce the impact of potential attacks.

Malik, R. et al. [11] propose an intrusion detection system utilizing the deep belief network (DBN). DBN is an algorithm that combines different unsupervised networks using autoencoders, specifically restricted Boltzmann machines (RBMs). The target classification of their model is either 0 (no intrusion detected) or 1 (intrusion detected). The TON_IOT dataset, consisting of 30,000 tuples, is employed for training and testing. This dataset is derived from a practical representation of a medium-scale network at the UNSW Canberra Cyber Range and IoT labs in Australia.

III. PROPOSED METHODOLOGY

This section provides a concise overview of the DEIIDS (Deep Learning & Elephant-based IoT Intrusion Detection System) methodology. Figure 1 illustrates the training model flow, detailing each block. Additionally, Figure 2 presents the block diagram outlining the testing process of the trained model. Table 1 enumerates various notations used in this work.

Table 1. BINIDS notation table.

Notations	Meaning
RID	Raw IOT Dataset
PID	Processed IOT Dataset
EHP	Elephant Population
n	Number of Elephant in BP
m	Number of Feature in PID
E_f	Elephant Fitness
BE_b	Best Fit ELEPHANT
FEF	Filter Elephant Features
Do	Desired Output

Pre-Processing of IOT Dataset

The dataset for the IoT network sessions contains both feature values and text collections. Some of these features exhibit repetition or constancy throughout the entire dataset. Therefore, a cleaning step in the algorithm involves removing these sets of features [12]. If RID represents the raw IoT dataset and PID represents the processed dataset, the cleaning process is denoted by the equation:

PID←IOTDatasetCleaning(RID)------Eq. 1

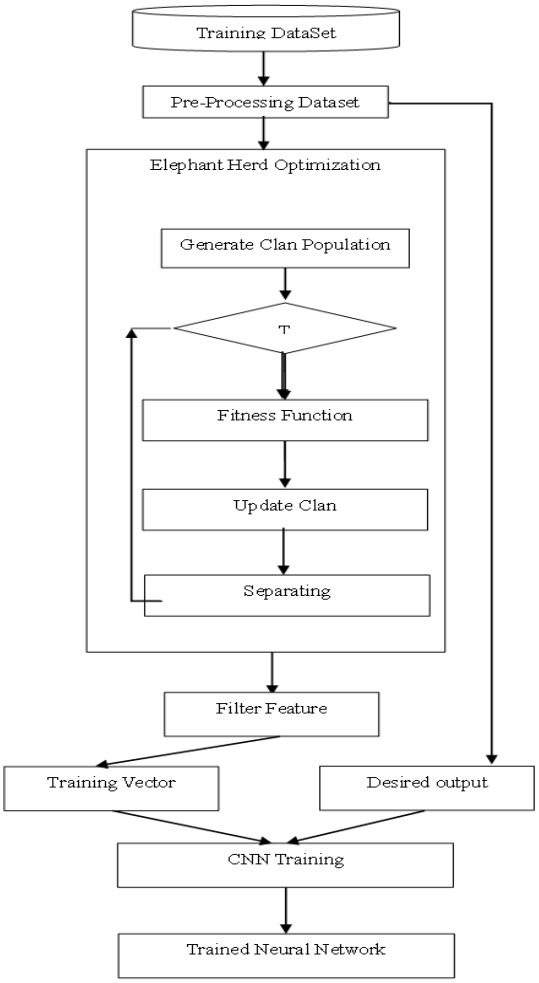

Fig. 1 Block diagram of proposed model.

Subsequently, the processed dataset undergoes normalization. This step is necessary because some feature values are within the range of 0 to 1000, while others are within 0 to 1 [13]. To standardize the scale, all values are transformed into a unified range of 0 to 1 by taking the ratio with others.

PID←Normalization(PND)----------Eq. 2

ElephantHERD Optimization

Elephants possess a distinct set of features determined by Gaussian noise. The clan population, denoted as matrix C, comprises n elephants, with each elephant having m potential features. Each elephant is represented by a set of binary values, where 1 indicates inclusion in the training

vector, and 0 denotes exclusion from training. Given the dynamic nature of genetic algorithms, feature positions are generated using a Gaussian function. Assuming the dataset contains m features in the PID matrix and b clans are generated, the Elephant Herd Population (EHP) is represented by an mxn matrix.

EHP←Generate_Herd(m, n)----------Eq. 3

Fitness Function

The fitness function is employed to rank each elephant based on the fitness value obtained during training and detection accuracy. Fitness values are evaluated by considering the distance. The Elephant's feature vector undergoes training and intrusion detection accuracy assessment within the neural network, serving as the fitness criterion for the population.

Ef←Fitness(EHP) ----------Eq. 3

Update Clan

A best solution matriarch, M is derived based on the fitness values of each elephant in clan in the population [14]. A number of the statuses were randomly changed based on the best matriarch, M feature set. The cloning is done by placing the best elephant set page in other elephant of clan.

C←Clan_update(EHP, BE)

Separating

Low fitness elephant were removed from the clan in form of male elephant [15]. This is done after estimating the new clan fitness value.

Filter Feature

Once iteration get complete then find best elephant from the last updated population. Feature having value one in chromosome consider as selected feature for training vector and other consider as unselected.

FEF←FilterFeatures(EHP, Bb)

Deep Neural Network

Convolution in CNNs involves applying a small convolution mask on the 2D input through the convolution operation, making CNNs easier to train and less prone to over fitting [16].

C←Convolution(EF, s, p, Fc)

Here, the stride (s) is a control variable determining the movement speed, with integer values. Padding (p) involves adding null rows or columns to the block if necessary. F represents the filter applied to the input EF.

Max-pooling

Max-Pooling, an essential concept, enlarges the receptive field by downsampling image feature maps in CNNs, commonly achieved through maximum or average pooling. After pooling the feature maps by a factor of a, the convolution operation becomes s times more effective in enlarging the receptive field. Typically, pooling operations use s = 2 to gradually encode high-level image features, and convolution and pooling operations often work together in groups.

C←Maxpooling (C, s, p, Fm)--------Eq. 8

Training of Convolutional Neural Networks

In the training of Convolutional Neural Networks, CNNs output the segmentation of the entire input image directly instead of computing the classes of individual pixels. The convolution and pooling operations preserve the image structure in hidden layers before fully connected operations, and these hidden feature maps contain complete information for every pixel in the image. The input block EF obtained after the convolutional operation serves as the training vector in CNN. The CNN is trained to predict whether the class is tumor or non-tumor. The trained CNN can then accept a processed (convolutional operation) blocked image as input and predict the block class (intrusion class or normal class).

DEIIDS← Train(FBF, Do)

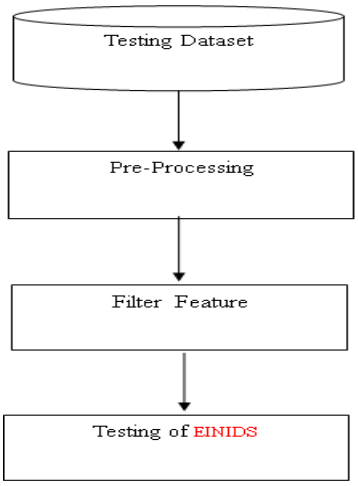

Fig. 2 Testing of BININDS model.

Proposed DEIIDS Algorithm
Input: RID

Output IDTNN

1. PID←IOTDatasetCleaning(RID)
2. PID←Normalization(PND)
3. EHP←Generate_Herd(m, n)
4. Loop 1:itr
5. E_f←Fitness(EHP)
6. C←Clan_update(EHP, BE)
7. EndLoop
8. E_f←Fitness(EHP)
9. BE←max(E_f)
10. TV←Training_Vector(BE_f, PID)
11. DO←Desired_Output(PID)
12. EndLoop
13. DEIIDS ← Train(TV, Do)

Testing of BINIDS Model

In order to test DEIIDSmodel testing dataset takes IOT dataset as input. Pre-processing steps are same as done in training phase. Further in this phase no need to run ELEPHANT algorithm, just filter selected feature as per ELEPHANT cluster centers obtained while training session. Finally pass the selected feature in trained DEIIDSmodel.

IV. EXPERIMENT AND RESULTS

Experimental setup: BINIDS and comparing model was developed on MATLAB software. Experimental machine having 4 GB ram, i3 6th generation processor. Comparison of BINIDS was done with previous IOT malicious session detection model proposed in [18]. : IOT dataset was taken from [19]. This dataset has 86 attributes, where three is the class of session and rest 83 is to training/testing features. Total number of sessions are 625784, with two and multiclass named sessions.

Evaluation Parameter

$$\text{Precision} = \frac{\text{True\_Positive}}{\text{True\_Positive} + \text{False\_Positive}}$$

$$\text{Recall} = \frac{\text{True\_Positive}}{\text{True\_Positive} + \text{False\_Negative}}$$

$$F\_Score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$Accuracy = \frac{Correct\_Classification}{Correct\_Classification + Incorrect\_Classification}$$

Results

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

Dataset Size	**Previous Work**	DEIIDS
5000	0.923	0.9772
10000	0.9984	0.976
15000	1	0.9722
20000	0.9236	0.9786
25000	0.8365	0.9824

Precision values of IOT network intrusion detection models shown in table 2. It was found that proposed modle has increases the true alarm of the detection. Further it was found that use of elephant herd optimization for the training feature optimization has increases the learning accuracy.

Table 3. Recall value-based comparison of network intrusion detection models.

Dataset Size	**Previsous Work**	DEIIDS
5000	0.799	0.975
10000	0.9346	0.962
15000	0.942	0.985
20000	0.7633	0.9908
25000	0.8654	0.993

Table 3 shows that recall values of the proposed model is high as compared to previous existing work. Further it was found that use Deep learning convolution operation on selected feature transform the values in more relevant values.

Table 4. F-Measure value-based comparison of network intrusion detection models.

Dataset Size	**Previous Work**	DEIIDS
5000	0.8568	0.976
10000	0.9655	0.969
15000	0.97	0.9786
20000	0.8359	0.9847
25000	0.8507	0.9878

F-measure values of table 4 shows that true alarm of intrusion class detection of proposed model is high. It was found that feature optimization and convitional transformation increases the learning of the model. So deep learning with feature optimization is effective to detect the IOT network intrusion.

Table 5. Accuracy value-based comparison of network intrusion detection models.

Dataset Size	**Previous Work**	DEIIDS
5000	89.53	97.721
10000	95.38	98.07
15000	97.33	97.7
20000	89.66	98
25000	91.62	98.25

Accuracy values of IOT network intrusion detection models shown in table 5. It was found that proposed model has increases the true alarm of the detection. Further it was found that use of elephant herd optimization for the training feature optimization has increases the learning accuracy.

Table 6. Accuracy of Normal intrusion class detection models.

Dataset Size	**Previous Work**	DEIIDS
5000	95.7067	97.7213
10000	99.676	97.6021
15000	100	97.2174
20000	96.6758	97.86
25000	93.553	98.2362

Table 7. Accuracy of DOS intrusion class detection models.

Dataset Size	**Previsous Work**	DEIIDS
6000	99	100
8000	99.8775	98.4483
12000	100	98.448
14000	98.4483	98.448
16000	97.06	98.448

Table 8. Accuracy of Probe intrusion class detection models.

Dataset Size	**Previous Work**	DEIIDS
5000	33.0078	89.0625
10000	32.696	89.2925
15000	41.8738	89.2925
20000	29.2543	89.2925
25000	27.533	89.2925

Table 9. Accuracy of R2L intrusion class detection models.

Dataset Size	**Previous Work**	DEIIDS
5000	98.2866	97.6636
10000	99.519	98.3180
15000	98.5595	98.3180
20000	98.3192	98.318
25000	99.319	98.318

Table 10. Accuracy of U2R intrusion class detection models.

Dataset Size	**Previsous Work**	DEIIDS
5000	93.6782	54.233
10000	94.0267	50.076
15000	94.0267	50.0761
20000	14.968	50.0761
25000	67.3226	50.0761

Table 6, 7, 8, and 9 shows that accuracy values of the proposed model is high to detect all class of intrusion as well. Further it was found that use Deep learning convolution operation on selected feature transform the values in more relevant values.

IV. CONCLUSİON

This paper has proposed IOT network security model that increases the life by alarming attack sessions. This paper has proposed a model that increases the work learning efficiency by reducing the feature set. It was found that use of elephant herd for optimization reduces the epoces of learning. Further it was found that convolutional and maxpooling operator works fine in all set of intrusion class detection. Result shows that proposed work has improved the detection accuracy by 5.24%. Precision value by 4.09%. In future scholar can perform experiment in under water IOT network environment.

REFERENCES

[1] Kyriazis D, Varvarigou T, White D, Rossi A, Cooper J. “Sustainable smart city IoT applications: Heat and electricity management & Eco-conscious cruise control for public transportation,”. IEEE 14th International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM). Madrid, Spain. 2013;2013:1–5.

[2] Saba T, Saba T, et al. Real-time anomalies detection in the crowd using convolutional extended short-term memory network. J Inform Sci. 2021.

[3] Schneider S. The industrial internet of things (iiot) applications and taxonomy. Internet Things Data Anal Handb. 2017;41–81.

[4] Alkahtani H, Aldhyani THH, Al-Yaari M. Adaptive anomaly detection framework model objects in cyberspace. Appl Bionics Biomech. 2020;6660489:14.

[5] Tangsatjatham P, Nupairoj N (2016) Hybrid big data architecture for high-speed log anomaly detection. In: 2016 13th International joint conference on computer science and software engineering (JCSSE), pp 1–6. IEEE

[6] Visa S, Ramsay B, Ralescu AL, Van Der Knaap E (2011) Confusion matrix-based feature selection. MAICS 710(1):120–127

[7] ASwedan, A. N. Khuffash, O. Othman, and A. Awad, “Detection and prevention of malicious cryptocurrency mining on internet-connected devices,” in Proceedings of the 2nd International Conference on Future Networks and Distributed Systems, pp. 1–10, Amman, Jordan, June 2018.

[8] Kumar R., Kumar P., Tripathi R., Gupta G. P., Garg S., & Hassan M. M. (2022). A distributed intrusion detection system to detect DDoS attacks in blockchain-enabled IoT network. Journal of Parallel and Distributed Computing, 164, 55–68.

[9] Le K. H., Nguyen M. H., Tran T. D., & Tran N. D. (2022). IMIDS: An intelligent intrusion detection system against cyber threats in IoT. Electronics, 11(4), 524.

[10] Nasr, T.; Torabi, S.; Bou-Harb, E.; Fachkha, C.; Assi, C. Power jacking your station: In-depth security analysis of electric vehicle charging station management systems. Comput. Secur. 2022, 112, 102511.

[11] Malik, R.; Singh, Y.; Sheikh, Z.A.; Anand, P.; Singh, P.K.; Workneh, T.C. An improved deep belief network ids on iot-based network for traffic systems. J. Adv. Transp. 2022, 2022, 7892130.

[12] S. Latif, Z. e. Huma, S. S. Jamal, F. Ahmed, J. Ahmad, A. Zahid, K. Dashtipour, M. Umar Aftab, M. Ahmad, Q. H. Abbasi, Intrusion Detection Framework for the Internet of Things using a Dense Random Neural Network, IEEE Transactions on Industrial Informatics (2021).

[13] S. W. Azumah, N. Elsayed, V. Adewopo, Z. S. Zaghloul, C. Li, A Deep LSTM based Approach for Intrusion Detection IoT Devices Network in Smart Home, in: 7th IEEE World Forum on Internet of Things, WF-IoT 2021, 2021, pp. 836–841.

[14] Li, J.; Guo, L.; Li, Y.; Liu, C. Enhancing Elephant Herding Optimization with Novel Individual Updating Strategies for Large-Scale Optimization Problems. Mathematics 2019.

[15] Ala Saleh Alluhaidan, "Secure Medical Data Model Using Integrated Transformed Paillier and KLEIN Algorithm Encryption Technique with Elephant Herd Optimization for Healthcare Applications", Journal of Healthcare Engineering, vol. 2022, Article ID 3991295, 14 pages, 2022.

[16] Bazgir, O., Zhang, R., Dhruba, S.R. et al. Representation of features as images with neighborhood dependencies for compatibility with convolutional neural networks. Nat Commun 11, 4391 2020.

[17] Fatani, M. AbdElaziz, A. Dahou, M. A. A. Al-Qaness and S. Lu, "IoT Intrusion Detection System Using Deep Learning and Enhanced Transient Search Optimization," in IEEE Access, vol. 9, pp. 123448-123464, 2021, doi: 10.1109/ACCESS.2021.3109081.

[18] 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.