

Honey Badger Algorithm-Based Hyperparameter Optimization of A Multi-Head Attention Network for IoT Intrusion Detection

P. Kavitha³

³Research Scholar, Department of Electrical and Electronics Engineering
Government College of Engineering, Tirunelveli, Tamilnadu, India - 627007. kavitha.paulsamy6@gmail.com

Corresponding Author E-mail

ABSTRACT: Internet of Things (IoT) devices have increased the complexity of network environments and exposed them to various cyber-attacks, for maintaining network security. However, Intrusion Detection (ID) techniques often struggle in handling high-dimensional network traffic data, identifying the most relevant features. To overcome these challenges, this paper proposes a Honey Badger Algorithm-Based Hyperparameter Optimization of a Multi-Head Attention Network (HBA-MHAN) for IoT-ID. Data pre-processing based on data cleaning and label encoding used to improve the quality then stability of the input data. Data visualization is used to analyse the features in addition distribution of the network traffic. Subsequently, Recursive Feature Elimination (RFE) is applied to select the most appropriate features and eliminate redundant informative attributes. The selected features are delivered to the MHAN, which learns important relationships among network traffic features through multiple attention heads. The HBA is employed to optimize the key hyperparameters of the MHAN, improving the learning process and ID performance. The optimized MHAN performs ID and classifies network traffic into the corresponding attack or normal categories. Experimental evaluation on the TON_IoT network dataset using Python demonstrates that the proposed HBA-MHAN achieves 95% accuracy, 96% precision, 98.55% recall, and 97% F1-score for IoT-ID. The proposed structure improves ID performance whereas reducing the impact of irrelevant and redundant network features for efficient and accurate IoT-ID.

Keywords: Internet of Things, Intrusion Detection, Honey Badger Algorithm, Multi-Head Attention Network.

1. Introduction

IoT is a network of interconnected physical devices, including household appliances, autonomous vehicles, wearable devices, and industrial machines [1]. These devices use embedded software and communication technologies to connect to the internet, allowing them to accumulate, share, in addition to interchange data with other devices and systems [2-3]. Consequently, they continuously monitor network traffic, identify unusual activities, detect and respond to emerging cyber threats [4].

Furthermore, huge volume of data generated within the IoT poses significant challenges for IDS, limiting their ability to effectively identify as well as react to cyber threats [5]. Moreover, many IoT devices have inadequate computational resources, containing constrained memory, handling power, then energy, it difficult for them to efficiently process, achieve the large quantities of data produced through related devices [6]. Research on ID [7] is used to improve detection performance using sophisticated algorithms, whereas often providing less attention to their implementation in real-world environments.

2. Related Work

For IoT ID several techniques are utilized using Machine Learning (ML) and Deep Learning (DL) methods. In [8] Benmalek et al (2025) have proposed Particle Swarm Optimization (PSO) with K-Nearest Neighbors (KNN) for IoT ID. PSO identify the most informative network traffic features and remove irrelevant features. However, PSO requires multiple candidate solutions and iterations, whereas KNN performs distance calculations during classification. Consequently, in [9] Aljehane et al (2024) have presented Golden Jackal Optimization-Assisted Deep Learning Intrusion Detection System for Network Security (GJO-DL-IDSNS). It handles high-dimensional and nonlinear intrusion pattern. Nevertheless, GJO become trapped around a suboptimal solution in complex search spaces. Additionally, in [10] Alkhonaini, et al (2025) have developed Sandpiper Optimization-based Hybrid Deep Learning Intrusion Detection (SPOHDL-ID) framework for identifying malicious activities and enhancing security in IoT platforms. Nonetheless, SPO process requires multiple iterations before obtaining suitable model parameters. Furthermore, in [11] Chandnani, et al (2025) have invented Federated Multi-Layered Deep Learning (Fed-MLDL) model for ID it incorporates FedRIME, a physics-inspired hyperparameter optimization (HPO) technique, to identify suitable model parameters and improve the detection of malicious network activities. However, network traffic distributions differ significantly among contributing clients. Moreover, in [12] Zayed, et al (2026) have presented Firefly Algorithm (FA) with Convolutional Neural Network–Recurrent Neural Network (CNN–RNN) architecture for effective IoT ID. Nevertheless, it requires sufficient and representative IoT traffic data to learn diverse attack patterns.

2.1 Problem Statement

Existing ID techniques face several challenges like high dimensionality and large volume of IoT

network traffic data increase the computational complexity of ID and reduce detection efficiency. To overcome these problems this paper proposes an HBA-MHAN for IoT ID.

2.2 Contribution of the work

Proposed MHAN is used to capture complex feature relationships in IoT network traffic and improve discrimination between normal and attack traffic. HBA used to optimize the hyperparameter of MHAN technique, improving the learning efficiency.

3. Proposed Methodology

For effective IoT-ID the proposed methodology introduces a HBA-MHAN. The data pre-processing based on data cleaning is performed to remove inconsistent, duplicate, missing, and irrelevant records, whereas label encoding converts categorical class labels into numerical representations. Subsequently, data visualization is conducted to examine the distribution, statistical characteristics, correlations, and underlying patterns of the network traffic. Following pre-processing and visualization, RFE is used to identify the most informative network traffic features whereas eliminating irrelevant and redundant attributes. The selected features are then provided to the MHAN, where multiple attention heads learn diverse relationships and dependencies among the network traffic features and generate discriminative feature representations. To improve the effectiveness of the MHAN, the HBA optimize MHAN hyperparameter optimization technique. HBA searches for an effective combination of MHAN hyperparameters by improving the model learning efficiency and classification capability. The optimized MHAN is used to classify IoT network traffic into the corresponding normal and attack categories.

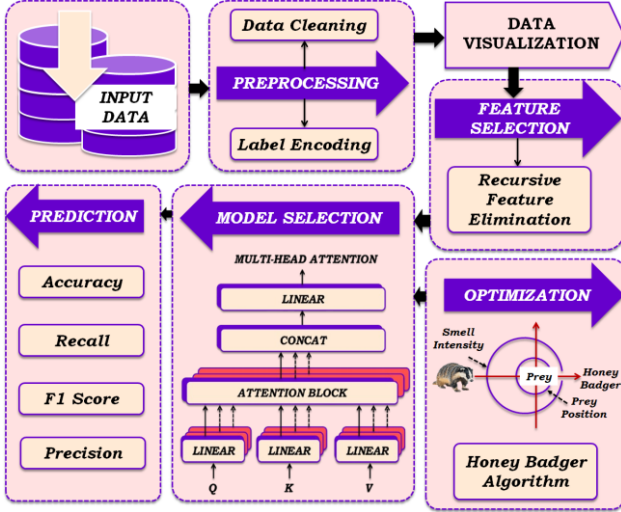


Figure 1: Proposed Block diagram for IoT-ID

3.1 Data Pre-processing

The raw TON_IoT network dataset is fed to data pre-processing based on data cleaning removes inconsistent values, irrelevant entries, and duplicate records. Label encoding is used to converts categorical class labels into numerical representations to improve data quality. The cleaned data is fed to data visualization for further analysis.

3.2 Data Visualization

For effective IoT-ID data visualization are performed to recognize the characteristics of IoT network traffic. Statistical distributions and relationships among network features are analysed using suitable plots like correlation heatmaps, feature-distribution visualizations. The data visualization identifies traffic patterns, feature relationships, potential class imbalance, and redundant attributes. The visualized data is fed to feature selection based on RFE.

3.3 Feature Selection

For IoT-ID RFE evaluates the contribution of each feature using a selected learning estimator. The least informative feature is recursively removed, and the remaining features are re-evaluated in consecutive iterations. RFE recollect traffic attributes that provide strong discriminatory information for distinguishing normal network

behaviour from malicious activities, whereas removing irrelevant and redundant features. These selected features are fed to model selection based on proposed MHAN.

3.4 Model Selection by Multi-Head Attention Network

For IoT-ID the proposed MHAN used to allocate dissimilar significance levels to network traffic features according to their contribution to ID. Figure 2 displays the design of MHAN.

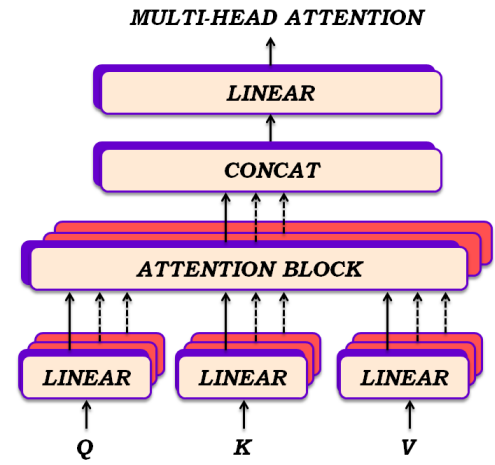


Figure 2: Construction of MHAN

MHAN mechanism consists of multiple self-attention operations, in every operation, the input is converted into three conditions the query (Q), key (K), as well as value (V) matrices. Here, the global feature vector matrix T , obtained from the multi-channel network, serves as the initial input for generating the matrices.

$$Q = K = V = T \quad (1)$$

The operation of Scaled Dot-product Attention (SDA) is expressed as

$$SDA(Q, K, V) = SoftMax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

The scaled by the $\sqrt{d_k}$ wherever d_k signifies the dimensionality of the key matrix K . The single-head attention mechanism first applies the softmax function to the similarity scores between the query vector and the key matrix to obtain the attention weights. These weights are then multiplied with

the value matrix to generate a weighted representation, effectively performing a weighted pooling operation over the value features for IoT-ID.

3.5 Honey Badger Algorithm

For IoT-ID the HBA optimize the MHAN parameters to improve the detection performance. HBA searches for the optimal solution by adjusting the target strength as well as applying two different hunting behaviours of HB the digging stage then the honey stage. Every stage follows a different search strategy to explore the solution space and improve the quality of candidate solutions. Figure 3 displays the flow chart for HBA.

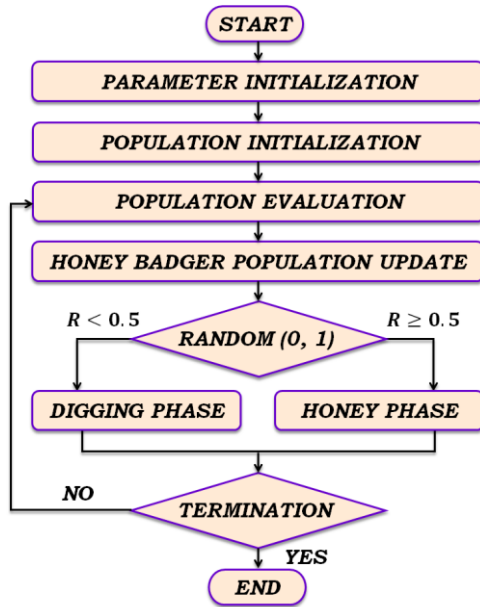


Figure 3: Flowchart for HBA

Through the initialization step, a populace of N HB is generated. The location of every HB represents a candidate solution and is initialized within the predefined search boundaries according to the following equation:

$$x_i = lb_i + r_1 \times (ub_i - lb_i) \quad (3)$$

Where, r_1 signifies a randomly generated value between 0 and 1, x_i denotes the position of the i^{th} HB and corresponds to a applicant resolution. The factors ub_i, lb_i term the lower and upper bounds

of the search space, respectively. The intensity is defined by

$$I_i = r_2 \times \frac{S}{4\pi d_i^2} \quad (4)$$

$$S = (x_i - x_{i+1})^2 \quad (5)$$

$$d_i = x_{prey} - x_i \quad (6)$$

$$\alpha = C \times \exp\left(\frac{-t}{t_{max}}\right) \quad (7)$$

Where, r_2 denotes a randomly generated value between 0 and 1, S indicates the target attention. The parameter d_i signifies the distance among the target as well as the first HB.

3.6 Digging stage

During this step, the HB uses its sense of fragrance to localize beehives then moves toward them to obtain food. Its movement follows a characteristic heart-shaped trajectory; the location of the HB is updated according to the subsequent calculation:

$$x_{new} = x_{prey} + F \times \beta \times I \times x_{prey} + F \times r_3 \times \alpha \times d_i \times |\cos(2\pi r_4) \times [1 - \cos(2\pi r_5)]| \quad (8)$$

$$F = \begin{cases} 1, & r_6 \leq 0.5 \\ -1, & r_6 > 0.5 \end{cases} \quad (9)$$

Where x_{prey} indicates the prey position, corresponding to the best position identified.

3.7 Honey Stage

In this step, the HB tracks the honey leader bird toward the hive to locate the honey source. This behaviour is mathematically denoted through the subsequent calculation:

$$x_{new} = x_{prey} + F \times r_7 \times \alpha \times d_i \quad (10)$$

Wherever, x_{new} signifies the updated location of the HB, x_{prey} designates the prey's position. The constraints F and α are calculated using Eqn. (7) and (9), correspondingly. By this the HBA finds the best solution for MHAN parameter in IoT-ID.

4. Result and Discussion

For IoT-ID the proposed HBA-MHAN is implemented in Python software using TON_IoT network dataset. With number of samples of

211043, number of features of 44, selected number of features of 20, the training samples of 152,379 and testing samples of 38095, the data is split as 80:20.

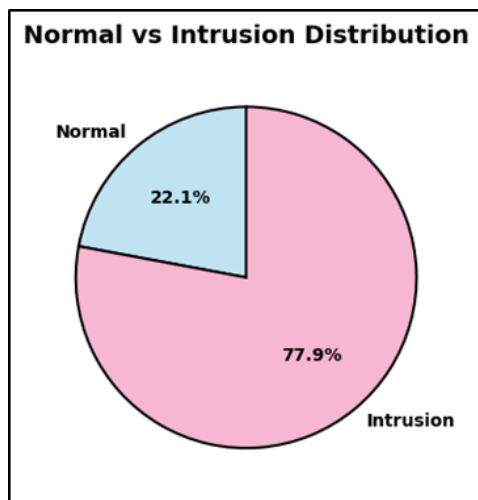


Figure 4: Normal vs intrusion distribution

Figure 4 displays the normal vs intrusion distribution. The normal have the value of 22.1% and intrusion distribution have the value of 77.9% for IoT-ID.

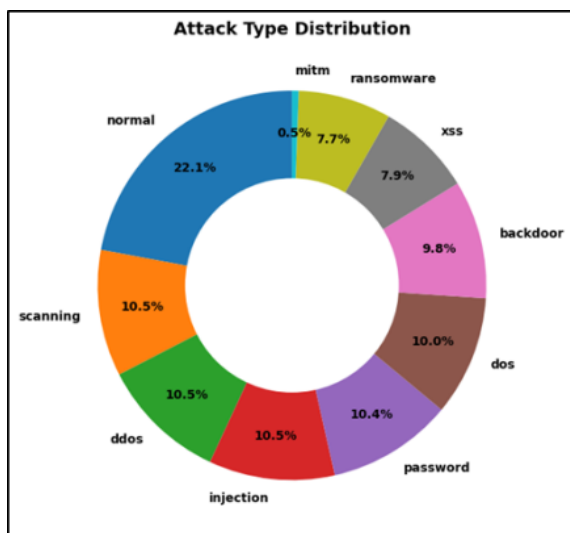


Figure 5: Attack type distribution

Figure 5 displays attack type distribution. It shows the percentages of various network attacks and normal traffic. Normal traffic have the value of 22.1%, scanning with 10.5%, ddos with 10.5%, injection with 10.5%, password with 10.4%, dos with 10.0%, backdoor with 9.8%, xss with 7.9%, ransomware with 7.7%, and mitm (man-in-the-

middle) with 0.5% it is low compared to other attacks.

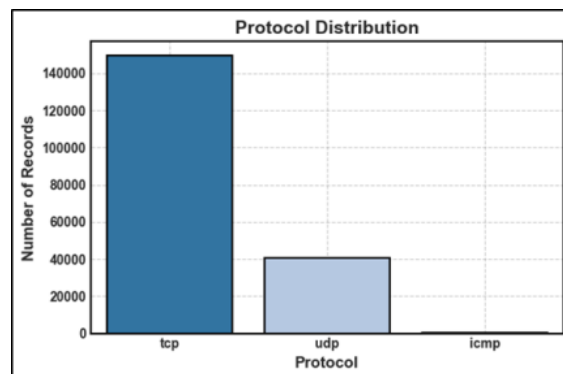


Figure 6: Protocol distribution

Figure 6 shows the protocol distribution. The network protocol used is tcp, udp and icmp, records up to 150000. The tcp is approximately up to 150,000 records which is the highest network, udp is approximately 40,000 records which have medium network and icmp is near 0 records which is the smallest network, indicating small network in IoT-ID.

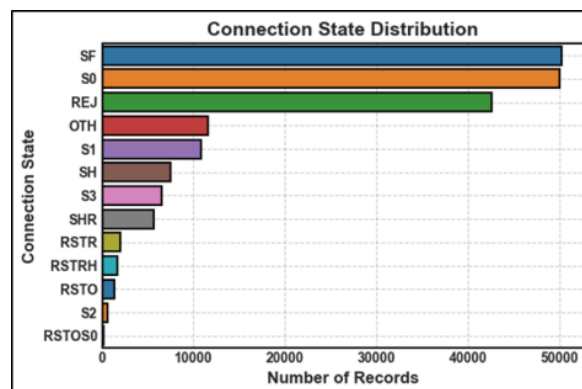


Figure 7: Connection state distribution

Figure 7 shows the connection state distribution. The network connection state SF is approximately 50,000 records showing normal connection establishment and termination. S0 is nearly 50,000 records showing the connection attempt seen with no reply. RSTR / RSTO indicating connection is reset by either the responder.

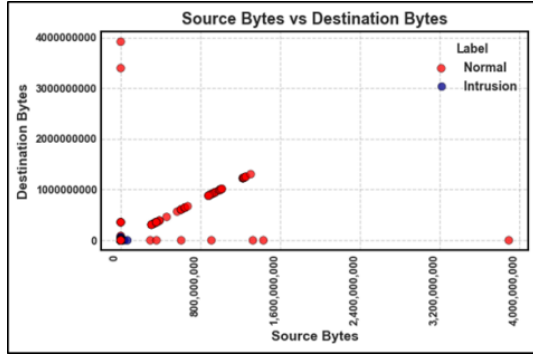


Figure 8: Source bytes vs distribution bytes

Figure 8 shows the source byte vs distribution bytes. The normal traffic is presented in red dots, concentrated mostly along a linear diagonal trend at lower-to-moderate values, with a few outliers at very high byte counts. Intrusion traffic is presented in blue dots, tightly clustered near the origin where both source and destination byte values are very small.

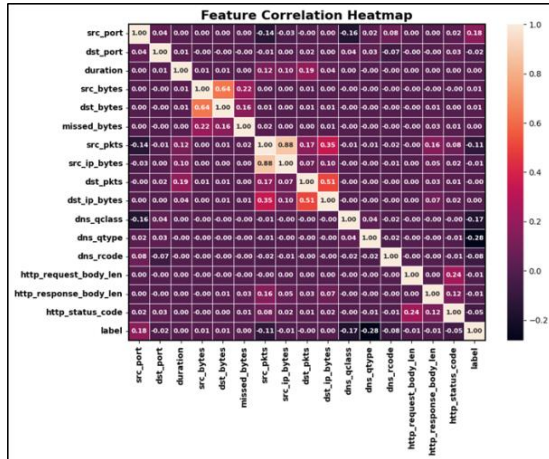


Figure 9: Feature correlation heatmap

Figure 9 displays the feature correlation heatmap. A network traffic dataset containing network and application layer features like ports, bytes, packets, and DNS/HTTP parameters for IoT-ID.

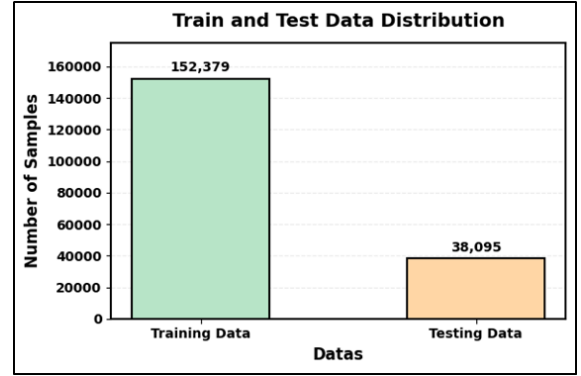


Figure 10: Train and test data distribution

Figure 10 shows the train as well as test data distribution. The number of samples for training the data is 152379 and number of samples for testing the data is 38095 for IoT-ID.

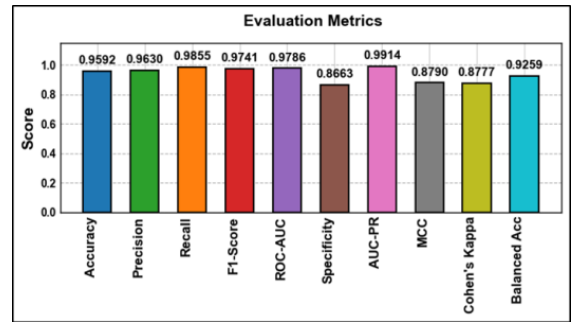


Figure 11: Evaluation Metrics

Figure 11 displays the evaluation metrics for IoT-ID using proposed HBA-MHAN. The proposed HBA-MHAN have the accuracy of 95.92%, precision of 96.3%, recall of 98.5%, F1-score of 97.4%, ROC-AUC of 97.8%, specificity of 86.6%, AUC-PR of 99.1%, MCC of 87.9%, Cohen's kappa of 87.7%, balanced accuracy of 92.5%, indicating good generalization.

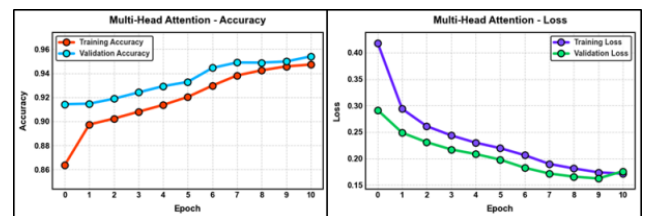


Figure 12: Multi-head Attention accuracy and loss

Figure 12 shows the MHA accuracy and loss. Both accuracy curves trend upward whereas both loss curves trend downward smoothly, demonstrating that the model is successfully learning the task.

The validation accuracy remains consistently equal to the training accuracy, and the validation loss tracks closely with the training loss. HBA-MHAN achieves stable training, indicates a well-regularized model with excellent generalization capability.

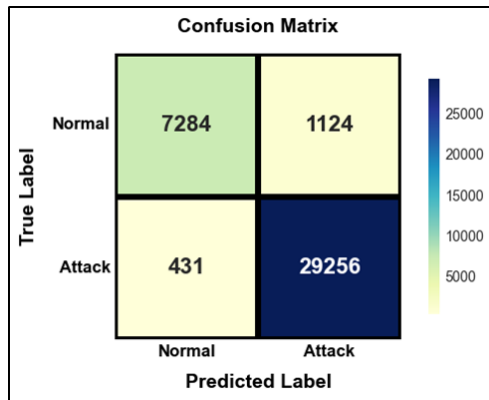


Figure 13: Confusion matrix

Figure 13 displays the confusion matrix for IoT-ID using proposed HBA-MHAN. The normal have the value of 7284, indicating stable and expected system operation, whereas the attack condition have a value of 29256, indicating a significant deviation from the normal operating condition.

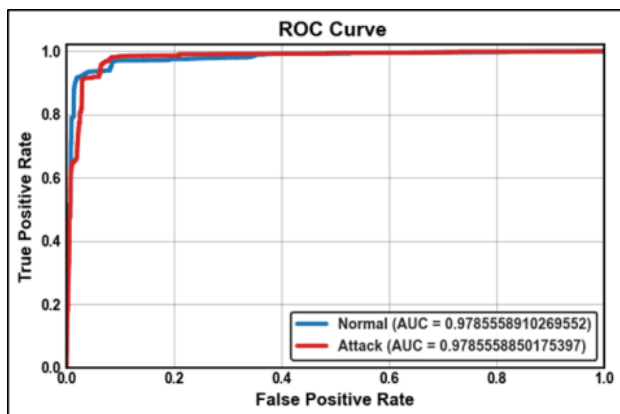


Figure 14: ROC curve

Figure 14 displays the ROC curve using proposed HBA-MHAN for IoT-ID. The normal and attack have the AUC of 0.978 indicating strong discrimination between normal and attack network traffic. The predicted class normal have the confidence of 99.9% and predicted class intrusions have the confidence of 100% for IoT-ID using proposed HBA-MHAN.

4.1 Comparison with the proposed technique

In the evaluation with proposed technique the proposed HBA-MHAN method is compared with the existing techniques as presented in table 1.

Table 1: Comparison with the proposed technique

Study	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Li et al [5]	87.04	85.65	86.78	86.14
Maseno et al [7]	92	83	98	90.6
Proposed Work	95.92	96	98.5	97

For IoT-ID Li et al [5] have the accuracy of 87.04%, precision of 86.65%, recall of 86.78%, F1-Score of 86.14%, Maseno et al [7] have the accuracy of 92%, precision of 83%, recall of 98% as well as F1-Score of 90.6% in addition proposed HBA-MHAN have the accuracy of 95.92%, precision of 96%, recall of 98.5% then F1-Score of 97%, indicating good performance compared to existing techniques.

5. Conclusion

In this paper HBA-MHAN is proposed for effective ID in IoT environments. Data pre-processing effectively improves the quality of IoT network traffic data and reduces the influence of redundant and irrelevant features, resulting in more reliable intrusion detection. The RFE effectively identified relevant traffic patterns and accurate discrimination between normal and attack activities. The MHAN, which captures important relationships among network traffic attributes, whereas the HBA optimizes the key hyperparameters of the network to enhance its classification performance. The proposed approach is evaluated using the TON_IoT network dataset in Python and achieved an accuracy of 95%, precision of 96%, recall of 98.55%, then F1-score of 97%. These outcomes validate the capability of the HBA-MHAN framework to

effectively distinguish normal and attack traffic whereas improving the utilization of high-dimensional IoT network data. The proposed approach delivers an effective framework for accurate and efficient IoT-ID and support the development of network security mechanisms for IoT environments. Future work focuses on lightweight attention-based architectures to reduce computational complexity and allow deployment on resource-constrained IoT and edge devices.

References

1. Muktar Bahruddin Panjaitan; Janaki Sivakumar; Archana Chabra; Menila James; Harpreet Kaur; Chetna Vaid Kwatra, Year: 2023, "Intrusion detection system based on machine learning models: An empirical analysis", In AIP Conference Proceedings, AIP Publishing LLC, Vol: 2930, No: 1, pp. 020040.
2. Arun Sekar Rajasekaran; S. Shanmuga Priya; Janaki Sivakumar; Akshay Varkale, Year: 2023, "Cybersecurity Measures Using Machine Learning for Business Applications", In 2023 4th International Conference on Computation, Automation and Knowledge Management (ICCAKM), IEEE, pp. 01-08.
3. M. Sivasubramanian; Kondalarao Punati; Jeya Bright Pankiraj; V. Usha Rani; V. Kiranmayee, Year: 2025, "Empowering DR Assessment: A DenseNet-CBAM Driven Approach", In 2025 3rd International Conference on Data Science and Information System (ICDSIS), IEEE, pp. 1-6.
4. , Deepesh M. Dhanvijay; Mrinai M. Dhanvijay; Vaishali H. Kamble, Year: 2025, "Cyber intrusion detection using ensemble of deep learning with prediction scoring based optimized feature sets for IOT networks", Cyber Security and Applications, Vol. 3, pp. 100088.
5. Jing Li; Mohd Shahizan Othman; Hewan Chen; Lizawati Mi Yusuf, Year: 2024, "Optimizing IoT intrusion detection system: feature selection versus feature extraction in machine learning", Journal of Big Data, Vol. 11, No. 1, pp. 36.
6. Abdallah Elayan; Michel Kadoch, Year: 2026, "Enhancing IoT network security: A BPSO-optimized attention-GRU deep learning framework for intrusion detection", Computers, Vol. 15, No. 5, pp. 266.
7. Elijah M. Maseno; Yanxia Sun; Zenghui Wang, Year: 2026, "GA-optimized deep learning intrusion detection framework with LIME explainability for IoT networks", Evolutionary Intelligence, Vol. 19, No. 1, pp. 23.
8. Mourad Benmalek; Abdessamed Seddiki, Year: 2025, "Particle swarm optimization-enhanced machine learning and deep learning techniques for Internet of Things intrusion detection", Data Science and Management, Vol. 8, pp. 423-435.
9. Nojood O. Aljehane; Hanan Abdullah Mengash; Majdy M. Eltahir; Faiz Abdullah Alotaibi; Sumayh S. Aljameel; Ayman Yafoz; Raed Alsini; Mohammed Assiri, Year: 2024, "Golden jackal optimization algorithm with deep learning assisted intrusion detection system for network security", Alexandria Engineering Journal, Vol. 86, pp. 415-424.
10. Mimouna Abdullah Alkhonaini; Manal Abdullah Alohal; Mohammed Aljebreen; Majdy M. Eltahir; Meshari H. Alanazi; Ayman Yafoz; Raed Alsini; Alaa O. Khadidos, Year: 2025, "Sandpiper optimization with hybrid deep learning model for blockchain-assisted intrusion detection in iot environment", Alexandria Engineering Journal, Vol. 112, pp. 49-62.

11. Chirag Jitendra Chandnani; Vedik Agarwal; Shlok Chetan Kulkarni; Aditya Aren; D. Geraldine Bessie Amali; Kathiravan Srinivasan, Year: 2025, “A physics-based hyper parameter optimized federated multi-layered deep learning model for intrusion detection in IoT networks”, IEEE Access, Vol. 13,pp. 21992-22010.
12. Samar M. Zayed; Samah Alshathri; Walid El-Shafa, Year: 2026, “Firefly algorithm optimized hybrid deep learning framework for intrusion detection in IoT environments”, International Journal of Computational Intelligence Systems, Vol. 19, No. 1, pp. 158.