Optimized DDoS Detection in Smart Homes Using EPSO and Recurrent Transformer Networks

Sanaa Ali Jabber

Faculty of Administration and Economics, Administration and Economics, AL-Muthanna University, Iraq.

Corresponding Author: Sana.ali@mu.edu.iq

Received 29 Mar. 2025, Accepted 30 Apr. 2025, Published 30 June. 2025.

DOI: 10.52113/2/12.01.2025/117-131

Abstract: The risk of Distributed Denial-of-Service (DDoS) attacks on smart home systems is increasing due to the advanced nature of network complexity and the rise in the application of encrypted information. To improve DDoS detection, this research suggests using a new framework that combines an Enhanced Particle Swarm Optimization (EPSO) with a Recurrent Transformer Network (RTN). The collection and examination of network data is done at the packet level as well as the flow level. Through analyzing this information, the EPSO algorithm can detect important patterns even when they are in low volume, while the RTN can track temporal patterns even in encrypted communication. Experimental evaluations demonstrate the framework's superiority over traditional methods, achieving higher accuracy, reduced false alarms, and improved smart home security against DDoS threats. Where we achieved an accuracy of 98%, recall of 99%, precision of 96%, F1-score of 97%, and an AUC of 99%.

***Keywords*:** DDoS Attacks, IoT Networks, Attack Detection, Encrypted Data, Smart Home Networks, Traffic Analysis

1 Introduction

Nowadays, Distributed Denial of Service (DDoS) attacks pose a great threat to network systems for which they use flooding traffic so that services fail and operations break down [1]. Most modern DDoS attacks are carried out through botnets, which enable an attacker to control a large number of infected devices and turn them into effective attack tools aimed at disrupting vital services and disrupting digital infrastructure [2, 3]. The Internet of Things-based smart home systems, especially those with devices such as cameras and smart appliances, have a higher risk because they use general security gaps such as the default passwords and un-updated security holes [4, 5, 6]. Attackers exploit these vulnerabilities to compromise the devices and install malware on them, making them part of a botnet [7, 8, 9]. Advancements made over time have seen researchers move away from studying elementary attack tools such as Trinoo and Tribe Flood Network (TFN) in the late 1990s [10, 11] to analyzing more sophisticated attack techniques such as Shaft and Stacheldraht in the early 2000s [12]. With the development of attack tools such as mstream and TFN2K [13, 14], it has become more complex for ones that capitalize on AI/ML in detecting as well as analyzing traffic for possible threats [15, 16, 17].

Lately, innovative techniques such as federated learning and Software Defined Networking (SDN) have been developed to

improve the capability of detecting and combating DDoS attacks targeting Internet of Things (IoT) environments to strengthen resilience and ensure better security posture of smart home systems [18, 19, 20].

2 Related works

This section provides an analysis of some key studies that have employed different models, data sets, and algorithms for improving DDoS attack detection efficacy in IoT networks.

In this research paper [21], a combination of Machine Learning (ML) and Deep Learning (DL) techniques was employed to identify DDoS attacks within IoT based on the CIC2023 IoT dataset.

Different types of algorithms, such as Logistic Regression, K-Nearest Neighbors (KNN) and Deep Neural Networks (DNN) were applied for efficient classification of DDoS attack signatures. The output was seen to have very high accuracy in detecting the attacks and lower false alarms, with precision, recall, and F1-score all above ≈0.9999. One recent study [25] introduced a security model known as Message-Driven Reinforcement Learning (MD-RL), which applies reinforcement learning techniques to protect edge computing environments within IoT networks. The model was implemented using the NS-2.35 simulator in a virtual network comprising resource-constrained IoT nodes and high-capacity edge nodes. To evaluate its effectiveness, the researchers simulated Distributed Denial-of-Service (DDoS) attacks targeting real-time communication between IoT devices and edge servers. The results demonstrated that the proposed MD-RL approach offered adaptive and efficient protection, outperforming conventional security methods in terms of both performance and cost. The study [26] proposed a model that uses Particle Swarm Optimization (PSO) and Artificial Neural Networks (ANN) to enhance the Intrusion Detection System (IDS) in Wireless Sensor Networks (WSNs). Monte Carlo simulation is employed to extract key features such as region size, detection range, and transmission distance. The optimized The PSO-ANN model achieves 90% detection accuracy, surpassing traditional methods like decision trees (DT) and naive Bayes (NB), which showed lower performance. The results also indicate improvements in classification accuracy and precision reducing Root Mean Square Error (RMSE). The study highlights the model's capability to enhance intrusion detection efficiency in networks with limited resources. The study [27] proposes a modified Particle Swarm Optimization (PSO) model to address the IP traceback (IPTBK) issue in Distributed Denial of Service (DDoS) attacks. The proposed model, PSO-IPTBK, utilises a Source Path Isolation Engine (SPIE) to reconstruct the attack path by analysing collected data packets. The goal is to trace the origin of the attack by reconstructing the most probable route between the attacker and the victim. Experimental results demonstrate that the model achieves a high accuracy of 98.33% in a 24-node network, outperforming other models in reducing the number of packets required for route reconstruction. OMNeT++ simulation and the INET 4 Framework were used to test the model's effectiveness. The results confirm the model’s high accuracy in identifying attack paths, thus enhancing DDoS detection and mitigation strategies. Study proposed an enhanced Intrusion Detection System (IDS) that utilises Grey Wolf Optimizer (GWO) combined with Long Short-Term Memory (LSTM)-basedrrent Neural Networks. The research addresses the challenge of

handling large feature sets in network data, which can degrade the IDS's performance in terms of accuracy and processing speed. By incorporating feature selection and classification techniques, the study improves the IDS's effectiveness in detecting cyberattacks.e proposed method is evaluated using NSL-KDD and UNSW-NB15 datasets, achieving 96.1% and 97.4% accuracy, respectively. The results indicate significant performance improvements compared to existing methods. The study [29] proposed a hybrid deep learning model, AE-MLP, for DDoS attack detection and classification. The model combines autoencoderE) for effective feature extraction and multilayer perceptronLP) for classifying DDoS attacks into specific types. The AE automatically identifies the most relevant features from large datasets, improving detection accuracy and reducing computational overhead. The model'sperformance, tested on the CICDDoS2019 dataset, shows high accuracy rates, surpassing 98% in attack detection and classification. The AE-MLP approach outperforms many similar methods in both precision and recall, proving to be an efficient solution for real-time DDoS defense.

The paper [30] proposed a data-mining-based DDoS attack prediction system for the IoT environment. The system consists of two key modules: first, the DDoS attack prediction model construction and second, the DDoS attack prediction defense module. In the first module, Support Vector Machine (SVM) is employed to classify and identify potential attacks, while continuously refining the prediction results. The study shows that the system can predict the timing of DDoS attacks, allowing for proactive defense using IP backtracking to trace and block attack sources in real-time, thus enhancing IoT security.

3 Operating Environment of the Proposed System

The proposed system for detecting DDoS attacks in smart homes is designed to operate within an IoT environment, where various smart devices such as security cameras, lighting controllers, voice assistants, and sensors are interconnected via a network This environment presents multiple challenges, including device diversity, the nature of streaming data, and the encryption that enhances the security of information shared through communication may also hinder the monitoring of malicious activities, hence complicating the detection process.

All these factors together result in a complicated environment where one cannot easily ensure that there are no vulnerabilities in IoT systems.

4 Components of the Operating Environment

This system is composed of many linked units that operate together efficiently for the purpose of analyzing network data and detecting assaults. Every module has its function towards improving how the whole system works.

Modules may encompass but are not limited to data collection, preprocessing data, extraction of characteristics, identification of anomalies, integration of threat intelligence, as well as response protocols. When put together, these modules offer a complete package that helps in recognizing and preventing internet hazards instantly, thereby guaranteeing strong network security.

A. IoT Devices:

- This category includes smart cameras, sensors, smart locks, smart refrigerators, and other connected devices within the home network.
- These devices communicate with each other via wired or wireless networks, making them vulnerable to cyberattacks, particularly DDoS attacks.

B. Router and Firewall:

- The router acts as a gateway between the home network and the internet, handling all incoming and outgoing data.
- A basic firewall is applied to filter suspicious data and block known threats; however, it is insufficient for detecting advanced attacks such as DDoS.

C. Data Collection and Analysis Unit:

This particular device works by collecting data from different points of a connection. It does so in two ways.

- Firstly, on the packet level, it can gather even the most detailed information contained in every packet that moves around in the network. Such information consists of key elements like source and destination addresses, protocols used, as well as packet length, among others.
- On a broader traffic level, the unit examines how different data flows between neighboring nodes in order to determine if there is any inappropriate traffic.

The collected information is then passed on for pre-processing, which involves cleaning and organizing so that it can be analyzed properly in the future phase.

D. Intelligent Analysis Unit Using EPSO and Recurrent Transformer Network:

This unit uses the Enhanced Particle Swarm Optimization (EPSO) algorithm. It works on optimizing vital parameters in attack identification. The output of the processed data goes to the Recurrent Transformer Network (RTN), which analyzes temporal patterns and recognizes anomalies in traffic.

Particularly effective and novel is this approach because it can analyze encrypted data, thereby identifying anomalies in traffic that may indicate arising DDoS attacks.

E. Response and Mitigation Unit:

As soon as an attack is identified, certain steps are taken to protect the network.

- All suspicious traffic is blocked to prevent the propagation of the attack on the network.
- Security alerts are sent out to users as well as network administrators so that they may take appropriate security measures.
- Extra actions: redirecting traffic toward safe servers and increasing firewall strength to improve security.

This approach focuses on the immediate identification and prevention of DDoS attacks in smart homes, thus enhancing overall network security and protecting devices from harm. Components of the operating environment for DDoS Detection System in Smart Homes in Figure 1.

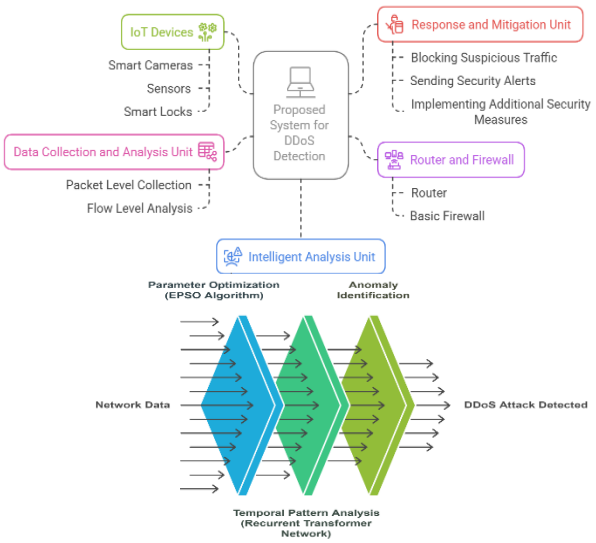

Fig. (1): Components of the Operating Environment for DDoS Detection System

5 Optimization the Parameters by EPSO

The system utilizes the Enhanced Particle Swarm Optimization (EPSO) algorithm at this point to adjust the needed parameters to detect any form of attack. With this optimization process, it becomes much easier to enhance the real-time DDoS attack detection performance level by increasing both the accuracy as well as making it work faster without any hitches; such features ensure that all menaces will be seen on time and handled properly by the system. The critical parameters requiring optimization can be categorized as follows:

5.1 Feature Selection

During the detection process, it is crucial to carry out feature selection since it aids in the identification of important attributes within the network traffic data. After all this is done, then it will be possible for the model to have some features that are important and therefore it will tell when there is an attack by comparing normal data traffic from the other one. The most significant features include:

- **Packet Size:** Variation in packet sizes can indicate the presence of an attack, as malicious traffic often exhibits distinct size patterns.
- **Flow Duration:** Short-lived, high-frequency traffic flows are characteristic of **DDoS** attacks.
- **Source/Destination IP Distribution:** Anomalies in the distribution of source or destination IP addresses may signify **botnet-driven** traffic.
- **Protocol Types:** The prevalence of certain protocols, such as **UDP, ICMP, or TCP SYN floods**, can indicate attack vectors.
- **Traffic Entropy:** Entropy measures deviations in network behavior, where low entropy values may indicate highly uniform, attack-generated traffic.

By optimizing feature selection, the system enhances its capacity to identify attack patterns while reducing computational overhead.

5.2 Threshold Values for Anomaly Detection

It is important to establish dynamic thresholds for key traffic features because they enable us to distinguish between normal changes and attack indications. When these thresholds are fine-tuned, the system is able to detect anomalous activities at very low levels of traffic, which in turn reduces false alarm rate but increases true positive alarms. With such precision in place, one can react better as events unfold. The primary thresholds include:

- **Traffic Volume Spikes:** A sudden increase in incoming requests may indicate a volumetric **DDoS attack**.
- **Flow Density Variations:** Unusual concentrations of traffic within short timeframes suggest an ongoing attack.

• **Packet Inter-Arrival Time:** Abnormal deviations in packet arrival times may indicate attack patterns, particularly in **low-rate DDoS attacks**.

The optimization of these thresholds allows the system to dynamically adjust to evolving attack strategies, ensuring **real-time** anomaly detection.

5.3 Hyperparameters of the Recurrent Transformer Network

A recurrent transformer network is employed by the system for an effective analysis of time-series network traffic data, which gives it the capacity to handle data that comes. The performance of this model depends on the fine-tuning of several hyperparameters including:

- **Learning Rate:** it determines the speed at which the model can learn or update itself. If not set well, a high learning rate may lead to unreliable training results, while a very low one could decelerate convergence unduly.

- **Number of Attention Heads:** Adding more attention heads improves pattern recognition by the model, particularly in complex network flows, thus enhancing overall detection accuracy.

- **Hidden Layer Units:** Determines the model's capacity to **capture long-term dependencies** in network traffic, which is essential for detecting persistent attacks.

The proper optimization of these parameters significantly enhances the model's robustness in detecting **DDoS** attacks within dynamic network environments.

6 Methodology

EPSO is employed to select the most relevant features from the raw network traffic data. The goal is to improve detection accuracy while reducing computational complexity.

1. A swarm of particles (candidate parameter sets) is randomly initialized.

Define a feature set $F = \{f_1, f_2, \dots, f_n\}$

Each particle represents a subset of features **encoded as a binary vector**

$X^t = (x_1^t, x_2^t, \dots\dots, x_n^t)$

where:

$$x_1^t = \begin{cases} 1 \; if \; feature \; f_i \; is \; selected. \\ 0 \; if \; feature \; f_i \; is \; \text{excluded.} \end{cases}$$

2. Evaluate the fitness of each particle using a pre-trained RTN model.

$$F(X) = \alpha \times \text{Detection Accuracy} + \beta \times (1 - \text{False Positives}) + \gamma \times \text{Detection Rate} \quad (1)$$

Where:
α, β, and γ are weighting factors to balance accuracy, false alarms, and Detection Rate, which can be adjusted according to the system's priorities (e.g., increasing the importance of accuracy or reducing cost).

3. Update velocities and positions using the EPSO update rules [21]:

$$v_i(t+1) = w \times v_i(t) + c_1 \times r_1 \times (p_{best} - x_i) + c_2 \times r_2 \times (g_{best} - x_i) \quad (2)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (3)$$

Here, p_{best} is the best position of the individual particle, g_{best} is the best global position found. w is the inertia weight, c_1, c_2 are acceleration coefficients, and r_1, r_2 are random values between 0 and 1.

4. Repeat the process until convergence criteria are met.

5. Select the optimal feature subset for DDoS detection.

//After selecting the best feature set, EPSO is used again to fine-tune the hyperparameters of the RTN model for improved detection accuracy.

1) Learning Rate η: Controls how fast the model updates during training.

2) Number of Attention Heads h: Determines how different attention mechanisms focus on different network traffic features.

3) Hidden Units u: Affects the ability to capture temporal dependencies.

4) Dropout Rate d: Helps prevent overfitting.

5) Batch Size B: Influences the training stability.

Encode each particle as a vector:

$$J(X) = \alpha \times Accuracy - \beta \times False\ Positives - \gamma \times Computational\ Cost \quad (4)$$

6. Apply EPSO to search for optimal hyperparameters using the same velocity and position update rules as in feature selection.

7. Train the RTN model using the best hyperparameters.

Algorithm (1) Hyperparameter Optimization Using (EPSO) with Dynamic Adjustment of Acceleration Coefficients and Inertia Weight

Input:

-Network traffic features

-Parameter ranges (learning rate, dropout rate, etc.)

Output:

-Optimized feature subset

-Best hyperparameters for the RTN model

1. PARAM_BOUNDS= {

Packet_Size ,[1500 ,64]

Flow_Duration ,[10000 ,1]

Packets_per_Flow ,[1000 ,1]

IP_Distribution,[1 ,0]

Protocol_Type [0, 3], // 0: TCP, 1: UDP, 2: ICMP, 3: Other

Traffic_Entropy,[1 ,0]

Anomaly_Threshold,[0.99 ,0.01]

Dynamic_Threshold_Adjustment,[1 ,0]

Sensitivity_Parameter,[1.0 ,0.1]

Learning_Rate,[0.01 ,0.0001]

Num_Attention_Heads,[8 ,2]

Hidden_Layer_Units,[256 ,16]

Batch_Size[128 ,16] }

2. initialize swarm with N particles

for each particle i:

initialize position $p_i = \{\lambda_1, \lambda_2 \ldots, \lambda_n\}$ randomly

initialize velocity $v_i = \{v_1, v_2 \ldots, v_n\}$ randomly

Set initial values for inertia weight `w`, acceleration coefficients `$c_1$` and `$c_2$`

c_1_max = 2.5, c_1_min = 0.5

c_2_max = 0.5, c_2_min = 2.5

w_max = 0.9, w_min = 0.4

t = 0 // current iteration

T = max - iterations

Evaluate the fitness of each particle using:

$F(p_i) = \alpha \times \text{Detection Accuracy} + \beta \times (1 - \text{False Positives}) + \gamma \times \text{Detection Rate}$

3. Iterate until convergence or maximum iterations:

while t < T:

for each particle i:

for each parameter j:

$v_{ij}(t+1) = w \times v_{ij}(t) + c_1 \times r_1 \times (p_{best(ij)} - p_{ij}) + c_2 \times r_2 \times (g_{best(j)} - p_{ij})$

$p_{ij}(t+1) = p_{ij}(t) + v_{ij}(t+1)$

4. Dynamically adjust inertia weight `w`:

$w = w\_max - ((w\_max - w\_min) \times t\ /T)$

5. Dynamically adjust acceleration coefficients `$c_1$` and `$c_2$`

$c_1 = c_1\_max - ((c_1\_max - c_1\_min) \times t/T)$
$c_2 = c_2\_min + ((c_2\_max - c_2\_min) \times t / T)$

6. Evaluate the fitness of the updated particle: evaluate fitness $F(p_i(t+1))$
7. Update `$p_{\text{best}}$` and `$g_{\text{best}}$` if needed:
   if $F(p_i(t+1)) > F(p_{\text{best\_i}})$,
   $update \quad p_{\text{best\_}} = p_i(t+1)$
   If $F(p_{best}) > F(g_{best})$,
   $update \quad g_{best} = p_{best}$.
8. t = t + 1
9. Return the optimized hyperparameters= g_{best}.

Algorithm (2) DDoS detection system by Recurrent Transformer Networks (RTN)

Input:
Preprocessed network traffic sequences
EPSO-optimized model parameters
Output:
Classification labels for traffic flows (benign or malicious)

```
1. Transformer-model=initialize transformer-model(optimized hyperparameters)
2. Initialize optimized feature selection parameters by EPSO
feature_selection_params = optimized_hyperparameters  [6] // Assume the first 6 parameters are
related to feature selection.
selected_features = select_features (network_data, feature_selection_params)
3. // Initialize Anomaly Detection Parameters using EPSO
anomaly_detection_params = optimized_hyperparameters[6]
anomaly_threshold = anomaly_detection_params[0]
dynamic_threshold = anomaly_detection_params[1]
sensitivity = anomaly_detection_params[2]
4. //Training a RTN using network data (selected features)
   train_network_with_data(transformer-model, selected-features)
5. //Perform analysis on live (streaming) data
predictions = make_predictions (transformer_model, live_network_traffic)
6. //Detect attacks using time patterns
for prediction in predictions:
  if prediction == 'DDoS'  //Classify the attack as DDoS
     trigger_mitigation_actions ()
notify_admin()
```

Figure 2 presents the overall architecture of the proposed framework for DDoS attack detection in smart home IoT environments. The system operates in two integrated stages: optimization using Enhanced Particle Swarm Optimization (EPSO) and detection using a Recurrent Transformer Network (RTN).

Stage 1: EPSO Optimization (Left Side of the Figure)

This stage begins by defining the search space for critical hyperparameters (such as learning rate, number of layers, dropout rate, etc.). A swarm of particles is initialized, each representing a candidate solution. EPSO evaluates the performance (fitness) of each particle by measuring the classification accuracy of the RTN model when trained with the respective parameter set.

The algorithm then iteratively updates particle positions and velocities to explore the solution space efficiently. Once the best-performing configuration is found (i.e., convergence is achieved or the maximum iteration limit is reached), EPSO returns the optimized hyperparameters for use in the RTN model.

Stage 2: RTN-Based DDoS Detection (Right Side of the Figure)

In this stage, the optimized hyperparameters are used to initialize the Recurrent Transformer Network. Relevant features are selected based

on prior analysis, and the RTN is trained using labeled network traffic data.

Once deployed, the RTN model continuously analyzes live network traffic and detects patterns that indicate potential DDoS attacks. The model leverages its self-attention mechanism to capture temporal dependencies, even within encrypted data streams. If an attack is detected, the system immediately triggers predefined mitigation actions and alerts the network administrator.

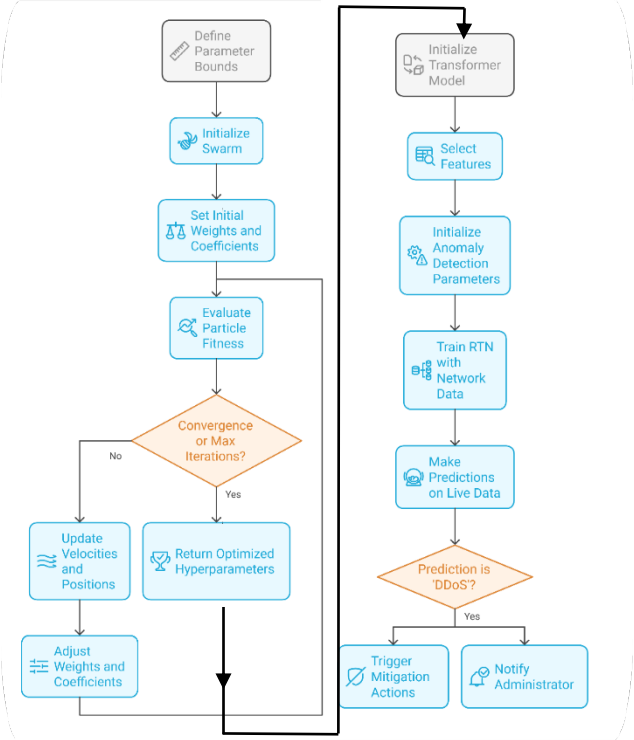

Fig. (2): DDoS detection system by EPSO +RTN

7 Performance Evaluation

The performance evaluation of the proposed DDoS detection system for smart home environments was conducted through a comprehensive set of experiments to assess its effectiveness and robustness. Important performance indicators, which include Accuracy, Recall, Precision, F1 Score, Area Under the Receiver Operating Characteristic Curve (AUC), and False Positive Rate (FPR) were used to evaluate the system, as shown in Table 1 and Table 2. In engaging in this, it is possible to determine whether or not the system can identify malicious traffic without too many false alarms using these parameters alone. Consequently, this critical review determines how well the identification process can protect from, or at least mitigate, DDoS attacks.

Table 1: Confusion Matrix Table [23, 24].

	Predicted Attack	**Predicted Normal**
Actual Attack	True Positive (TP)	False Negative (FN)
Actual Normal	False Positive (FP)	True Negative (TN)

- **True Positive (TP)**: The number of cases where an attack was correctly detected.
- **False Positive (FP)**: The number of cases where normal traffic was mistakenly classified as an attack.
- **False Negative (FN)**: The number of cases where the system failed to detect an attack.
- **True Negative (TN)**: The number of cases where normal traffic was correctly classified.

Table 2: Basic Performance Metrics Table.

Metric	**Formula**
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$
False Positive Rate (FPR)	$\frac{FP}{FP+TN}$

F1 Score	$2 \times \frac{Precision \times Recall}{Precision + Recall}$
Specificity	$\frac{TN}{TN + FP}$
AUC (Area Under ROC Curve)	Value between 0 and 1 (The closer to 1, the better the performance)

7.1 The Impact of Threshold Value on the Performance of the Proposed System

In order to enhance the effectiveness of DDoS detection mechanisms in smart homes, it had to first establish how changes in threshold values may influence their efficacy. The aim was to determine the perfect threshold that strikes a balance between Recall and Precision because they are very crucial in improving model accuracy in a real-world setting.

The effectiveness of the system was analyzed by taking different threshold values from 0.1 to 0.4 and finding out the ones that give better performance concerning certain key metrics: Accuracy, Recall, Precision, F1 Score. The study found that with an increase in the threshold, there was an improvement in the accuracy of the system, as shown in Table 3 and Figure (3). This was possible because, at a threshold value of 0.4, the system could strike a trade-off between precision and recall, hence obtaining the highest F1 score.

Table 3: The Impact of Different Threshold Values on the Performance of the Proposed System.

Threshold	Accuracy	Recall	Precision	F1 Score
0.1	0.94	0.97	0.90	0.93
0.2	0.95	0.96	0.92	0.94
0.3	0.96	0.95	0.93	0.94
0.4	0.97	0.94	0.94	0.95

Analysis results suggest that a threshold of 0.4 gives the proposed DDoS detection system its optimal performance. The F1 Score is maximum when the threshold is set at this point; this means that there is a good trade-off between sensitivity and false alarm probability. In order for the smart home network to remain operational and reduce the harmful consequences of false alerts and incorrect classifications on overall network resilience, it is important to strike this balance by all means.

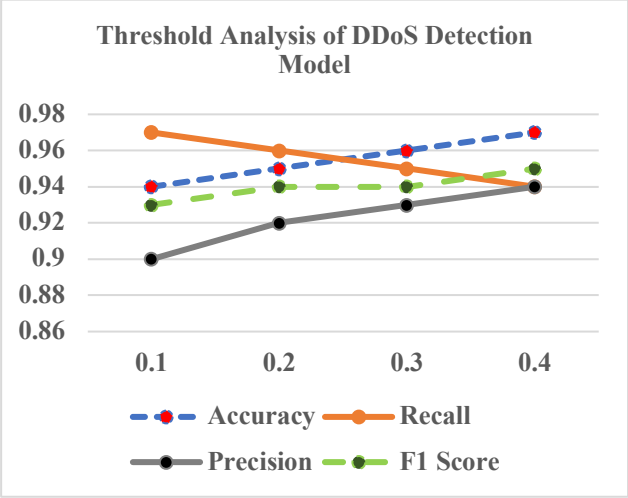

Fig. (3): The Impact of Different Threshold Values on the Performance of the Proposed System

7.2 Comparative Analysis of Model Performance

In addition to evaluating the impact of the threshold, a comparative analysis was performed between the proposed system, Random Forest, Artificial Neural Networks (ANN), and Support Vector Machine (SVM) models.

As illustrated in Table (4) and Figure (4), the study found that the EPSO + RTN system can

be more effective than standard machine learning techniques in identifying DDoS attacks on IoT smart homes. An accuracy level of 97%, F1 score = 0.96, and AUC = 0.98. Although the Artificial Neural Network (ANN) model has an impressive accuracy of 96%, it lacks the adaptability provided by the EPSO algorithm, which is critical for real-time tuning in a dynamic IoT environment.

Table 4: Models Comparison.

Model	Accuracy	F1 Score	AUC
Random Forest	95%	0.94	0.94
SVM	94%	0.93	0.93
ANN	%96	0.95	0.96
EPSO +RTN	**97%**	**0.96**	**0.98**

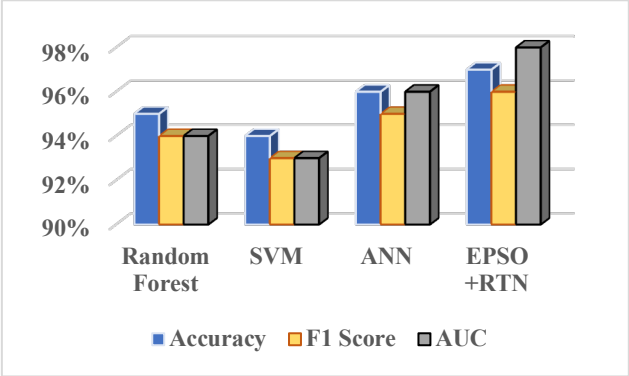

Fig. (4): Performance comparison between the proposed system, Random Forest, and Support Vector Machine (SVM) models.

comparative evaluation has been performed of the effectiveness of the proposed system with several existing methods, as summarized in Table (5) and Figure (5). Unlike previous approaches—such as MD-RL (2024), PSO + ANN (2023), and GWO-LSTM (2022)—which were primarily designed for IoT intrusion detection or WSN security, additionally, the integration of Enhanced Particle Swarm Optimization (EPSO) optimizes feature selection, while the Recurrent Transformer Network (RTN) effectively captures temporal patterns in network traffic.

One major benefit of the EPSO + RTN system is that it can handle encrypted traffic without having to decrypt it first, and as such, it does not interfere with the privacy of the users. It also boasts the lowest false positive rate (FPR), which is 2%, a factor that contributes to increased reliability through minimizing false alarms.

Table 5: Comparison table between the proposed system and previous works.

Study / Model	Techniques	Application Domain	Encrypted Traffic Analysis	Accuracy	Recall	Precision	F1 Score
Proposed System	EPSO + RTN	DDoS Attack Detection in IoT	✅	**98%**	**99%**	**96%**	**97%**
(2024),[25]	MD-RL	IoT-Cyber Attack Detection	❌	**98%**	**99%**	**96%**	**97%**
(2023),[26]	PSO + ANN	Intrusion Detection in WSN	❌	**96%**	**95%**	**92%**	**94%**

Study / Model	Techniques	Application Domain	Encrypted Traffic Analysis	Accuracy	Recall	Precision	F1 Score
(2019),[27]	PSO + IP	DDoS Path Tracing	✗	98.33% (small scale)	-	-	-
(2022),[28]	GWO + LSTM	Intrusion Detection in	✗	97.4%	-	-	-

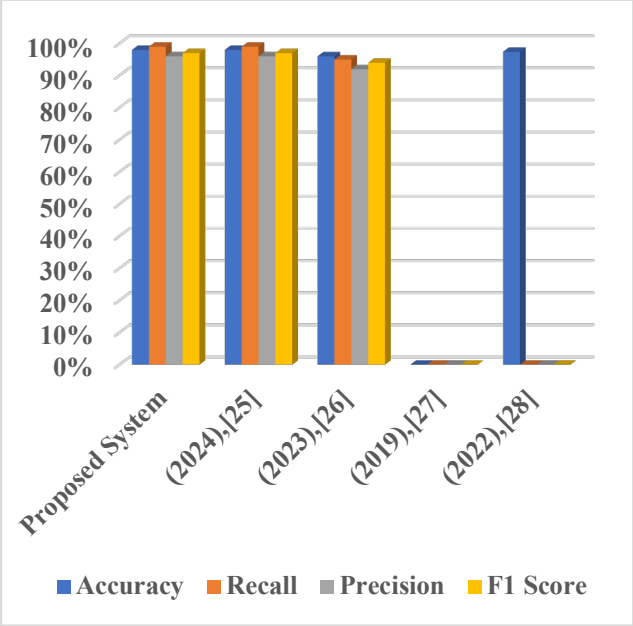

Fig. (5): Comparison between the proposed system and previous works.

8 Conclusion

The proposed DDoS detection system uses Enhanced Particle Swarm Optimization (EPSO) to optimize hyperparameters and employs Recurrent Transformer Networks (RTN) for time series attack detection. It improves precision and recall, such as precision and recall, while at the same time making sure that its false alarm rate is significantly low, so that it can offer better detection capabilities. The comprehensive experimental results confirm the system's capability to detect complex and distributed DDoS attacks effectively while ensuring minimal disruption to legitimate network activity. The integration of EPSO and RTN not only enhances detection accuracy but also improves adaptability and scalability, making the proposed solution a practical and reliable choice for securing smart home networks against evolving cyber threats. Furthermore, the ability to analyze encrypted traffic without decryption offers a significant advantage over prior models, ensuring robust security and privacy in modern IoT environments.

9 Limitations and Future Work

The EPSO + Recurrent Transformer Network (RTN) framework has been seen to potentially identify DDoS attacks in smart home IoT environment, several limitations remain that offer directions for future enhancement. Although with some few areas left where improvement may be applied.

1.Computational Overhead

Even though the model is optimized using EPSO to reduce computational cost, it may still be difficult to deploy it in real-time on IoT devices which have limited resources due to issues such as memory, latency, and energy

consumption. One possible solution could be developing a less complex model that can operate at the edge efficiently and with fewer resources.

2.Encrypted Traffic Diversity

the research investigated encrypted data flows, it could not distinguish among encryption protocols and traffic hiding methods. Further studies could determine the effect of different standard-based encryption (e. G., TLS 1. 3, VPN tunneling) on detection efficacy.

3. Absence of Adaptive Retraining Mechanism

The current model does not take into account continuous learning or retraining to keep up with the changing strategies of DDoS attacks.

Future Research Directions

- Development of a lightweight version of the RTN model suitable for edge devices.

- Testing the system with heterogeneous IoT devices and traffic patterns in real-time simulation platforms.

- Investigation of federated learning to preserve data privacy across multiple smart home networks.

References

[1] Alahmadi, A.A., Aljabri, M.,Alhaidari, F., Alharthi, D.J., Rayani,G.E., Marghalani, L.A., Alotaibi, O.B.,Bajandouh, and S.A., 2023, DDoS Attack Detection in IoT-Based Networks Using Machine Learning Models: A Survey and Research Directions. Electronics, 12, 3103. https://doi.org/10.3390/electronics12143103.

[2] Tuan, T. A., Long, H. V., Son, L. H., Kumar, R., Priyadarshini, I., and Son, N. T. K., 2020, Performance evaluation of Botnet DDoS attack detection using machine learning. Evolutionary Intelligence, 13(2), 283-294. https://doi.org/10.1007/s12065-019-00310-w.

[3] Lysenko, S., Bobrovnikova, K., Matiukh, S., Hurman, I., and Savenko, O., 2020, Detection of the botnets' low-rate DDoS attacks based on self-similarity. International Journal of Electrical and Computer Engineering, 10(4), 3651-3659. https://doi.org/10.11591/ijece.v10i4.pp3651-3659.

[4] Saxena, U., Sodhi, J. S., and Singh, Y., 2020, January. An analysis of DDoS attacks in a smart home networks. In 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 272-276). IEEE. https://doi.org/10.1109/Confluence47617.2020.9058087.

[5] Garba, U. H., Toosi, A. N., Pasha, M. F., and Khan, S., 2024, SDN-based detection and mitigation of DDoS attacks on smart homes. Computer Communications, 221, 29-41. https://doi.org/10.1016/j.comcom.2024.04.001.

[6] Ibrahim, M. I., and Darus, M. Y., 2024, DDOS Attack Analysis on IoT Device for Smart Home Environment and A Proposed Detection Technique. JOIV: International Journal on Informatics Visualization, 8(4), 2104-2110. https://doi.org/10.62527/joiv.8.4.2175.

[7] Raja, T. V., Ezziane, Z., He, J., Ma, X., and Kazaure, A. W. Z., 2022, October, Detection of DDoS Attack on Smart Home Infrastructure Using Artificial Intelligence Models. In 2022 International.Conference on Cyber-Enabled

Distributed Computing and Knowledge Discovery (CyberC) (pp. 12-18). IEEE. https://doi.org/10.1109/CyberC55534.2022.00014.

[8] Chandak, A. V., and Ray, N. K., 2024, DDoS attack detection in smart home applications. Software: Practice and Experience, 54(10), 2086-2101. https://doi.org/10.1002/spe.3249.

[9] Kalpana, A., and Waoo, A. A., 2023, IoT based smart home cyber-attack detection and defense. TIJER-Int. Res. J., 10(8), 16. https://ssrn.com/abstract=4537209.

[10] CERT Coordination Center, Nov. 1999, Distributed system intruder tools workshop report, Available: http://www.cert.org/reports/dsit_workshop.pdf .

[11] CERT Advisory CA-99-17, Sep. 1999, Denial-of-Service Tools, Available: http://www.cert.org/advisories/CA-99-17-denial-of-service-tools.html.

[12] Bashaiwth, A., Binsalleeh, H., and AsSadhan, B., 2023, An explanation of the LSTM model used for DDoS attacks classification. Applied Sciences, 13(15), 8820. https://doi.org/10.3390/app13158820

[13] D. Dittrich, G. Weaver, S. Dietrich, and N. Long, May 2000, The 'mstream' distributed denial of service attack tool, University of Washington, Available: https://www.cs.unc.edu/~jeffay/courses/nidsS05/attacks/mstream.analysis.txt.

[14] Singh, N., 2014, A Study on Cooperative Defense Against Network Attacks. Cosmos Journal of Engineering and Technology, 4(2), 1-4. https://www.cosmosjournal.in/wp-content/uploads/2020/10/CET-429.pdf.

[15] Dietrich, S., and Long, N., 2000, Analyzing distributed denial of service tools: The shaft case. In 14th Systems Administration Conference (LISA 2000).

[16] Moore, D., Shannon, C., Brown, D. J., Voelker, G. M., and Savage, S., 2006, Inferring internet denial-of-service activity. ACM Transactions on Computer Systems (TOCS), 24(2), 115-139.

[17] Almorabea, O. M., Khanzada, T. J. S., Aslam, M. A., Hendi, F. A., and Almorabea, A. M., 2023, IoT Network-Based Intrusion Detection Framework: A Solution to Process Ping Floods Originating from Embedded Devices. IEEE access, 11, 119118-119145. https://doi.org/10.1109/ACCESS.2023.3327061.

[18] Jabber, S. A., and Jafer, S. H., 2023, A novel approach to intrusion-detection system: combining LSTM and the snake algorithm. Jordanian Journal of Computers and Information Technology, 9(4). https://www.jjcit.org/paper/209/A-NOVEL-APPROACH-TO-INTRUSION-DETECTION-SYSTEM-COMBINING-LSTM-AND-THE-SNAKEALGORITHM.

[19] PM, V. P., and Soumya, S., 2024, Advancements in Anomaly Detection Techniques in Network Traffic: The Role of Artificial Intelligence and Machine Learning. Journal of Scientific Research and Technology, 38-48. https://doi.org/10.61808/jsrt114.

[20] Sáez-de-Cámara, X., Flores, J. L., Arellano, C., Urbieta, A., and Zurutuza, U., 2023, Clustered federated learning architecture for network anomaly detection in large scale

heterogeneous IoT networks. Computers & Security, 131, 103299. https://doi.org/10.1016/j.cose.2023.103299.

[21] Berqia, A., Bouijij, H., Merimi, A., & Ouaggane, A., 2024. Detecting DDoS Attacks using Machine Learning in IoT Environment. 1–8. https://doi.org/10.1109/iscv60512.2024.10620122

[22] Ali, Q. A., Elsakka, M. M., Korovkin, N. V., & Refaat, A. 2024. A novel EPSO algorithm based on shifted sigmoid function parameters for maximizing the energy yield from photovoltaic arrays: An experimental investigation. Results in Engineering, 24, 102967. https://doi.org/10.1016/j.rineng.2024.102967.

[23] Ahmad, R., Salahuddin, H., Rehman, A. U., Rehman, A., Shafiq, M. U., Tahir, M. A., and Afzal, M. S., 2024, Enhancing database security through AI-based intrusion detection system. Journal of Computing & Biomedical Informatics, 7(02).

[24] Anjimoon, M. S., Rahman, M. A., Vignesh, E., and Sampath, C., 2024, DISTRIBUTED DENIAL OF SERVICE ATTACK DETECTION USING RANDOM FOREST ALGORITHM IN MACHINE LEARNING. International Journal of Information Technology and Computer Engineering, 12(2), 201-211.

[25] Kumar, Anit, and Dhanpratap Singh, 2024, Detection and prevention of DDoS attacks on edge computing of IoT devices through reinforcement learning, International Journal of Information Technology 16.3, 1365-1376. https://doi.org/10.1007/s41870-023-01508-z.

[26] Narayanan, S. L., Kasiselvanathan, M., Gurumoorthy, K. B., and Kiruthika, V., 2023, Particle swarm optimization based artificial neural network (PSO-ANN) model for effective k-barrier count intrusion detection system in WSN. Measurement: Sensors, 29, 100875. https://doi.org/10.1016/j.measen.2023.100875.

[27] Lin, H. C., Wang, P., & Lin, W. H., 2019, Implementation of a PSO-based security defense mechanism for tracing the sources of DDoS attacks. Computers, 8(4), 88. https://doi.org/10.3390/computers8040088.

[28] Karthic, S., Manoj Kumar, S., & Senthil Prakash, P. N., 2022, Grey wolf-based feature reduction for intrusion detection in WSN using LSTM. International Journal of Information Technology, 14(7), 3719-3724. https://doi.org/10.1007/s41870-022-01015-7.

[29] Wei, Y., Jang-Jaccard, J., Sabrina, F., Singh, A., Xu, W., & Camtepe, S., 2021. Ae-mlp: A hybrid deep learning approach for ddos detection and classification. IEEE Access, 9, 146810-146821 https://doi.org/10.1109/ISCV60512.2024.10620122.

[30] Huang, L., 2022. Design of an IoT DDoS attack prediction system based on data mining technology. The Journal of Supercomputing, 78(4), 4601-4623. https://doi.org/10.1007/s11227-021-04055-1.