Unified Anomaly Detection in IoT and Cyber-Physical Networks Using Evo-Transformer-LSTM: Validation on Four CIC Benchmarks

Pardis Sadatian Moghaddam , Mahyar Mahmoudi , Nuria Serrano , Francisco Hernando-Gallego , Diego Martín *

Posted Date: 9 December 2025

doi: 10.20944/preprints202512.0763.v1

Keywords: Internet of Things; cyber-physical systems; intrusion detection datasets; Transformer; long short-term memory; chimp optimization algorithm

 doi:10.20944/preprints202512.0763.v1

Article

Unified Anomaly Detection in IoT and Cyber-Physical Networks Using Evo-Transformer-LSTM: Validation on Four CIC Benchmarks

Pardis Sadatian Moghaddam [1], Mahyar Mahmoudi [2], Nuria Serrano [3], Francisco Hernando-Gallego [4] and Diego Martín [3,*]

[1] Department of Computer Science, Georgia State University, Atlanta, Georgia, 30302, United States
[2] Faculty School of Industrial Engineering and Management, Oklahoma State University, Stillwater, OK, 74078
[3] Department of Computer Science, Escuela de Ingeniería Informática de Segovia, Universidad de Valladolid, Segovia, Spain
[4] Department of Applied Mathematics, Escuela de Ingeniería Informática de Segovia, Universidad de Valladolid, Segovia, Spain
* Correspondence: diego.martin.andres@uva.es

Abstract: The rapid proliferation of the Internet of Things (IoT) and cyber-physical systems (CPS) within critical infrastructure sectors has significantly expanded the attack surface for advanced and stealthy cyber threats. Since these systems increasingly rely on real-time data exchange and autonomous control, developing intelligent, scalable, and adaptive anomaly detection mechanisms has become a pressing requirement. This paper proposes a novel hybrid framework, evolutionary-transformer-long short-term memory (Evo-Transformer-LSTM), that integrates the temporal modeling capability of LSTM networks, the global attention mechanism of Transformer encoders, and the optimization power of the improved chimp optimization algorithm (IChOA) for hyper-parameter tuning. In the proposed architecture, the Transformer encoder extracts high-level contextual patterns from traffic sequences, while the LSTM component captures local temporal dependencies. The framework is rigorously evaluated on four benchmark datasets from the Canadian Institute for Cybersecurity (CIC): CIC-IDS-2017, CSE-CIC-IDS-2018, CIC IoT-DIAD (2024), and CICIoV (2024). Comparative experiments are conducted against several state-of-the-art baselines, including transformer, LSTM, bidirectional encoder representations from transformers (BERT), deep reinforcement learning (DRL), convolutional neural network (CNN), k-nearest neighbors (KNN), and random forest (RF) classifiers. Results show that the proposed Evo-Transformer-LSTM achieves up to 98.25% accuracy, an F1-score of 97.91%, and an area under the curve (AUC) of 99.36% on CIC-IDS 2017, while maintaining above 96% accuracy and 98% AUC even on the more challenging CICIoV 2024 dataset, consistently surpassing all baseline models. In addition, statistical significance tests confirm the superiority of the proposed approach. In conclusion, Evo-Transformer-LSTM offers a unified, scalable, and robust solution for anomaly detection in modern IoT and CPS infrastructures, with potential for real-world deployment in security-sensitive domains.

Keywords: Internet of Things; cyber-physical systems; intrusion detection datasets; Transformer; long short-term memory; chimp optimization algorithm

1. Introduction

The Internet of Things (IoT) has evolved into a pervasive ecosystem of interconnected electronic devices, encompassing sensors, actuators, embedded controllers, and edge computing units that collaboratively support intelligent decision-making [1-3]. These devices underpin a diverse range of

 doi:10.20944/preprints202512.0763.v1

applications, including precision agriculture, industrial automation, wireless power transfer, healthcare monitoring, smart grids, vehicular technology, energy management, and connected mobility [4-6]. Recent industry forecasts highlight the unprecedented scale of this growth: the global number of IoT connections reached 18.8 billion in 2024, representing a 13% increase over the previous year, and is projected to exceed 40 billion by 2030. Cellular IoT connections alone are expected to expand from 4 billion in 2024 to more than 7 billion by 2030, while short-range technologies such as Wi-Fi and Bluetooth will drive an even larger share of device proliferation, rising from 14.4 billion to approximately 35 billion connections during the same period [7-10]. This explosive expansion introduces unprecedented demands on traffic management and reliable connectivity, as IoT networks must accommodate massive volumes of heterogeneous data streams across dynamic and resource-constrained environments [11-13].

The exponential increase in device connectivity and heterogeneous traffic flows has also amplified the attack surface of IoT networks, exposing them to a wide spectrum of security threats. Massive volumes of unstructured, bursty, and latency-sensitive data streams make it difficult to ensure confidentiality, integrity, and availability at scale, leaving critical infrastructures vulnerable to malicious exploitation [14]. While IoT security spans multiple layers, ranging from cryptographic authentication and access control in higher-layer protocols [15-17] to physical layer security techniques for safeguarding wireless links [18-21], and down to hardware-level protections against side-channel attacks [22-24], one of the most persistent threats arises from network intrusions [25-28]. Sophisticated intrusion attempts can exploit weakly protected nodes, compromised firmware, or unmonitored communication channels, leading to service disruption, unauthorized data access, or even large-scale system failures. In this context, intrusion detection systems (IDS) have become indispensable as a second line of defense, capable of monitoring real-time traffic patterns and flagging abnormal behaviors that may bypass traditional preventive measures [29]. Among IDS approaches, anomaly detection plays a particularly vital role in IoT environments due to the dynamic and evolving nature of device behaviors, where predefined signatures of attacks are often insufficient [30]. By learning to distinguish normal operational patterns from subtle deviations, anomaly-based IDS provides the adaptability needed to safeguard next-generation IoT and CPS infrastructures.

Nowadays, traditional signature- or rule-based intrusion detection techniques struggle to cope with the scale, heterogeneity, and dynamic nature of IoT traffic, where attack patterns evolve rapidly and often lack predefined signatures [26]. In this context, deep learning (DL) has emerged as a promising paradigm for anomaly detection, owing to its ability to automatically extract hierarchical spatio-temporal features from complex and high-dimensional traffic data. Unlike conventional methods, DL-based models can adaptively capture hidden correlations and subtle deviations in real-time sequences, enabling the detection of novel or stealthy attacks that are otherwise difficult to identify. This adaptability, combined with the scalability of modern neural architectures, makes DL a powerful solution to enhance the robustness and generalization of intrusion detection systems across diverse IoT and CPS environments [30].

1.1. Related Works

CIC-based datasets such as CIC-IDS-2017, CSE-CIC-IDS2018, and their successors have become pivotal benchmarks for evaluating anomaly detection models, due to their diverse attack scenarios, structured flow-level features, and availability of raw traffic [13]. Building on these datasets, a growing body of research has proposed machine learning (ML) and DL frameworks tailored for identifying intrusions across various network environments [14,15]. This section provides a structured overview of such works, highlighting advances in feature selection, model architectures, hybrid and ensemble learning approaches, and dataset refinement. Special attention is given to methods addressing dataset imbalance, zero-day detection, encrypted traffic analysis, and deployment-oriented considerations such as inference speed and feature reduction.

A comprehensive analysis of benchmark datasets used in network intrusion detection systems (NIDS) was presented in [16], where the authors dissected the characteristics, structures, and relevance of eight major datasets, including CIC-IDS-2017 and CSE-CIC-IDS-2018. This foundational

 doi:10.20944/preprints202512.0763.v1

review helps researchers select suitable benchmarks for evaluating detection models. To address anomaly detection in encrypted network traffic, a hybrid DL model integrating convolutional neural networks (CNN) and gated recurrent units (GRU) was proposed in [17]. The framework, validated on NSL-KDD, UNSW-NB15, and CIC-IDS-2017, demonstrated high efficacy in extracting temporal features from encrypted traffic. In [18], three bio-inspired feature selection algorithms, artificial bee colony, flower pollination algorithm, and ant colony optimization, were compared on the CSE-CIC-IDS2018 dataset. Their approach optimized detection performance while reducing model complexity, achieving a near 99% accuracy.

The DOC-IDS framework proposed in [19] combined one-dimensional CNNs and autoencoders within a deep one-class classification strategy, enabling automatic feature extraction and enhanced anomaly detection. Trained on both regular and labeled open traffic, the model improved its discrimination capability with minimal manual effort. A hybrid detection model incorporating GRU into variational autoencoders, alongside a temporal correlation index (TCI), was introduced in [20] to improve anomaly detection in sequential network traffic. The system, tested on CIC-IDS-2017 and CIC-IDS-2018, significantly reduced false positive rates by up to 14.1%. Security in CI/CD cloud environments was enhanced in [21] by combining CNN and long short-term memory (LSTM) models to detect anomalies in pipeline traffic. Validated using both CSE-CIC-IDS2017 and 2018 datasets, the model reached over 98% accuracy, demonstrating effectiveness in securing software deployment pipelines. Despite its popularity, the CIC-IDS-2017 dataset was scrutinized in [22], where the authors revealed critical flaws in its data collection and labeling processes. By reconstructing and relabeling a significant portion of the dataset, they demonstrated the importance of dataset quality on model reliability.

In [23], deep LSTM models trained on the CSE-CIC-IDS2018 dataset achieved up to 99% detection accuracy. The study emphasized the applicability of DL to intrusion detection in the heterogeneous and high-volume context of IoT networks. Inspired by the human immune system, the work in [24] combined multiresolution wavelet analysis with a segmented deterministic Dendritic Cell Algorithm. Tested on multiple benchmarks including CIC-IDS-2017 and CSE-CIC-IDS2018, the model consistently reached near-perfect classification accuracy, especially outperforming alternatives on recent datasets. The authors in [25] introduced a novel CNN-based framework for improving intrusion detection on CIC-IDS datasets. By leveraging deep autoencoders for feature extraction and RF for preprocessing, the system attained F1-scores as high as 98.7% on CIC-IDS-2017 and 99.5% on CIC-IDS-2018.

A DL-based study in [26] evaluated six neural network models, including CNN, RNN, and hybrid CNN+LSTM, on the CSE-CIC-IDS2018 dataset, achieving over 98% classification accuracy in multi-class intrusion detection tasks. The study highlights the trade-off between model complexity and inference time for real-world deployment. In [27], the CIC-IDS2017 dataset was critically reviewed to understand its limitations in capturing modern network threats. The authors emphasized the need to develop more representative and diverse datasets to enhance the realism and effectiveness of future IDS models. The authors of [28] developed a dynamic anomaly detection system using LSTM enhanced with an attention mechanism and SMOTE to address class imbalance in the CSE-CIC-IDS2018 dataset. The model achieved 96.2% accuracy, demonstrating the importance of tailored loss functions and resampling techniques.

While most public IDS datasets are criticized for lacking realism, [29] introduced and visualized the CIRA-CIC-DoHBrw-2020 dataset using graph-based and statistical techniques. The dataset reflects modern encrypted traffic threats, offering high fidelity to real-world scenarios with imbalanced, low-footprint attack traits. An improved version of the CIC-IDS2017 dataset, named LYCOS-IDS2017, was proposed in [30] using a new tool called LycoSTand for better feature extraction. Performance comparisons showed enhanced results across multiple ML models, highlighting the benefits of refined dataset construction. The ICS-Flow dataset introduced in [31] focused on anomaly detection in ICS. It includes raw packets, flow records, and process variables, enabling supervised and unsupervised ML training. Several classifiers were validated, confirming their utility for ICS-oriented IDS research.

 doi:10.20944/preprints202512.0763.v1

To detect advanced persistent threats (APT), [32] proposed a hybrid ensemble model combining random forest (RF) and extreme gradient boosting (XGBoost). The system was validated on multiple benchmarks, including CSE-CIC-IDS2018 and CIC-IDS2017, reaching up to 99.91% accuracy with low false positive rates. A CNN-based U-Net and temporal convolutional network (TCN) were compared in [33] for anomaly detection on the CSE-CIC-IDS2018 and KDD99 datasets. The TCN+LSTM model showed superior performance, especially when trained with focal loss to handle time-series class imbalance. Feature selection using Information Gain on the CICIDS2017 dataset was studied in [34]. Models like RF and J48 reached an accuracy above 99.8%, showing that proper feature reduction improves both efficiency and performance in large-scale traffic analysis.

In [35], errors in the CIC-IDS2017 and CSE-CIC-IDS2018 datasets were systematically identified, including labeling and attack orchestration faults. The authors reconstructed a corrected dataset and emphasized the importance of transparent labeling logic for future benchmarking. The benchmarking study in [36] compared 31 ML models, including supervised and unsupervised algorithms, on the CICIDS2017 dataset. Results showed that k-NN, decision tree, and naive Bayes offered consistent high performance in anomaly detection with practical efficiency. The work in [37] applied a stacked autoencoder with RF and a multilayer perceptron on the CSE-CIC-IDS2018 dataset. Through correlation-based feature reduction, the proposed anomaly detection system effectively reduced detection time while maintaining high accuracy. Finally, [38] proposed a One-Class SVM framework combined with active learning for detecting known and unknown threats. Evaluated on CIC-IDS2017, the method outperformed conventional baselines, demonstrating its strength in handling zero-day attacks without prior labels.

1.2. Paper Motivation, Contribution, and Organization

The widespread adoption of CIC-based datasets has catalyzed substantial progress in network anomaly detection using ML and DL. However, a critical limitation persists: many existing models are trained and validated on a single dataset, making them highly dataset-specific and vulnerable to overfitting. This practice undermines the generalizability of the proposed methods across different network environments and traffic characteristics. Moreover, despite the availability of newer datasets, such as CICIoT2024 and CICIoV2024, that capture more recent and heterogeneous attack vectors, these resources remain underexplored in unified, multi-dataset evaluations. Additionally, while various hybrid architectures have been proposed, few efforts have explicitly integrated evolutionary optimization to jointly tune deep temporal architectures like Transformer and LSTM. This gap becomes particularly relevant in detecting subtle and stealthy intrusions that manifest differently across protocols and datasets.

To bridge these gaps, this paper proposes a unified anomaly detection framework that synergistically combines a Transformer encoder and an LSTM network, optimized via an Evolutionary algorithm. The model is evaluated across four diverse CIC-based benchmarks, CIC-IDS2017, CSE-CIC-IDS2018, CICIoT2024, and CICIoV2024, to assess its generalization capability across both traditional and next-generation IoT and vehicular traffic. By integrating temporal modeling, global attention, and adaptive hyper-parameter tuning within a single architecture, the proposed method aims to improve detection accuracy, reduce false positives, and maintain stability across heterogeneous traffic patterns and attack types. The main contributions of this work are summarized as follows:

- Unified DL Framework: We develop a novel evolutionary-Transformer-LSTM (Evo-Transformer-LSTM) architecture that integrates Transformer encoders with LSTM networks, enabling both long-term dependency capture and global temporal attention for robust anomaly detection.
- Task-Driven Hyper-parameter Optimization: We introduce a novel evolutionary hyper-parameter adjustment strategy tailored for hybrid deep architectures, enabling fine-tuning of both the Transformer and LSTM components to enhance detection accuracy, convergence behavior, and adaptability across varying network traffic patterns.

 doi:10.20944/preprints202512.0763.v1

- Cross-Dataset Validation: The model is comprehensively evaluated on four CIC-based datasets, namely CIC-IDS2017, CSE-CIC-IDS2018, CICIoT2024, and CICIoV2024, covering diverse application domains, including traditional networks, IoT, and vehicular traffic.
- Comprehensive Performance Analysis: Extensive experiments demonstrate the superiority of the proposed framework across accuracy, F1-score, and inference efficiency metrics, along with an analysis of its resilience to dataset imbalance and feature redundancy.

The remainder of this paper is organized as follows. Section 2 presents the materials and methods, including the benchmark datasets, the LSTM and Transformer modules, the improved chimp optimization algorithm (IChOA) optimizer, and the integration of these components into the proposed Evo-Transformer-LSTM framework. Section 3 outlines the experimental setup and presents the results obtained across four CIC-based benchmark datasets. Section 3 also discusses the key findings, highlighting the model's performance, stability, and generalization capability, with statistical validation and runtime analysis. Finally, Section 5 concludes the paper and outlines future research directions.

2. Materials and Proposed Methods

This section first introduces the benchmark datasets that capture diverse attack scenarios across enterprise networks, IoT environments, and vehicular systems, providing a broad experimental ground for evaluating the robustness of the proposed model. Beyond dataset description, this section emphasizes the methodological design, covering both DL modules and optimization strategies that together form a unified anomaly detection pipeline. To ensure clarity and reproducibility, the section is structured to progressively describe the core building blocks of the model. It begins with the baseline learning component, the LSTM, which captures temporal dependencies, followed by the Transformer encoder, which models global contextual relations. Subsequently, the IChOA is presented as the evolutionary optimizer for hyper-parameter tuning. Finally, the section integrates these components into the proposed Evo-Transformer-LSTM framework, highlighting how data flows through the architecture from raw input to optimized anomaly classification.

2.1. Dataset

In this paper, four benchmark datasets from the Canadian Institute for Cybersecurity (CIC) are employed to evaluate the proposed Evo-Transformer-LSTM framework: CIC-IDS-2017, CSE-CIC-IDS-2018, CIC IoT-DIAD (2024), and CICIoV (2024). These datasets were selected because they represent diverse real-world attack scenarios across traditional network intrusion detection, IoT traffic, and vehicular communication environments. Using multiple benchmarks ensures that the proposed deep model is not biased toward a single domain and can generalize effectively across heterogeneous IoT and CPS infrastructures.

- CIC-IDS-2017 is a widely used intrusion detection dataset that provides a comprehensive representation of modern attack vectors in conventional and IoT-enabled networks. It contains benign traffic as well as multiple categories of malicious activities such as brute force, botnet, denial of service (DoS), distributed denial of service (DDoS), infiltration, and web attacks. The dataset was generated using realistic network topology and traffic captures over a period of seven days, ensuring diversity and richness of attack behavior. The dataset includes more than 80 network flow features extracted using CICFlowMeter, covering packet statistics, flow duration, inter-arrival times, and byte counts. Its class distribution allows testing models against both balanced and imbalanced attack scenarios. Due to its richness and wide adoption in the research community, CIC-IDS-2017 serves as a strong baseline for validating anomaly detection approaches;
- CSE-CIC-IDS-2018 extends the earlier dataset by capturing a broader and more recent set of attack behaviors, offering an updated benchmark for evaluating anomaly detection systems. It includes benign traffic alongside advanced persistent threats such as brute force, SQL injection, infiltration, DDoS using botnets, and cryptomining attacks. The traffic was captured over several days from a realistic corporate-style environment with multiple services and user behaviors.

 doi:10.20944/preprints202512.0763.v1

This dataset also contains flow-based features extracted via CICFlowMeter, with more than 80 attributes per record. Importantly, its class structure includes both simple binary classification (benign vs. malicious) and multi-class categorization, enabling researchers to evaluate detection performance under different task formulations. By incorporating a wider range of up-to-date attack vectors, CSE-CIC-IDS-2018 ensures that models are assessed under more realistic and evolving threat conditions;

- CIC IoT-DIAD 2024 is a recently released dataset specifically designed for anomaly detection in IoT environments. Unlike the previous datasets that primarily represent enterprise traffic, this dataset emphasizes IoT device communication and traffic patterns, making it directly relevant to CPS and smart environments. It includes benign IoT traffic along with attacks such as scanning, man-in-the-middle (MITM), data injection, and denial of service. The dataset provides packet- and flow-level features that capture IoT-specific characteristics, such as device identifiers, lightweight protocol attributes (e.g., MQTT, CoAP), and timing irregularities. It contains multiple classes of benign and malicious traffic, allowing fine-grained evaluation of anomaly detection models in IoT contexts. Its importance lies in capturing the heterogeneity of IoT communication, which is often more lightweight and irregular compared to traditional IT traffic;
- CICIoV 2024 is a benchmark dataset for intrusion detection in vehicular networks, designed to capture traffic patterns in Internet of Vehicles (IoV) scenarios. It contains benign vehicular communication data as well as multiple types of attacks targeting vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. Examples include spoofing, jamming, denial of service, and false data injection. The dataset is constructed from realistic vehicular communication scenarios and includes both control-plane and data-plane traffic. The dataset provides detailed feature sets including packet timing, transmission intervals, payload content, and vehicular mobility attributes. It supports both binary classification and multi-class anomaly detection tasks. By including IoV-specific features, CICIoV 2024 ensures that the evaluation of anomaly detection frameworks extends into transportation systems, which are a critical component of CPS.

To ensure consistency across the four CIC benchmarks, all datasets underwent a systematic preprocessing pipeline before being used for training and evaluation. Since the datasets originate from different domains (enterprise traffic, IoT communication, and vehicular networks) the preprocessing phase was designed to unify their formats, reduce redundancy, and prepare the data for integration into the Evo-Transformer-LSTM framework. The main steps included cleaning and filtering, handling missing values, normalization of numerical features, and encoding of categorical attributes. For CIC-IDS-2017 and CSE-CIC-IDS-2018 datasets, which contain flow-level features extracted with CICFlowMeter, preprocessing began with the removal of duplicate records and incomplete flows to ensure data integrity. Missing values were imputed where necessary, and categorical features such as protocol type or service were encoded into numerical representations. Given the large number of statistical flow features, all numerical attributes were normalized into a fixed range to reduce scale bias between features like packet size and flow duration.

For CIC IoT-DIAD 2024 dataset, which emphasizes lightweight IoT traffic patterns, preprocessing involved careful treatment of device-specific attributes and protocol identifiers. Features extracted from IoT protocols such as MQTT and CoAP were retained, while redundant identifiers that did not contribute to classification were removed. Class balancing was also applied to mitigate the strong imbalance between benign device traffic and attack traces, ensuring that the model would not become biased toward majority classes. Normalization was then applied consistently to align with the other datasets. For CICIoV 2024, preprocessing accounted for vehicular communication attributes such as transmission intervals, payload length, and mobility-related parameters. Data cleaning steps included filtering corrupted packets and invalid timestamps, followed by normalization of all continuous features. Special attention was given to encoding categorical vehicular identifiers into fixed-length numeric formats, ensuring compatibility with the Transformer-LSTM input structure. These steps guaranteed that the vehicular dataset could be seamlessly integrated into the same feature space as the IoT and enterprise traffic datasets.

 doi:10.20944/preprints202512.0763.v1

2.2. LSTM

LSTM networks were introduced by Hochreiter and Schmidhuber in 1997 to address the limitations of traditional recurrent neural networks (RNNs), particularly the vanishing and exploding gradient problems that hinder their ability to capture long-term dependencies [39]. By incorporating a memory cell and a gating mechanism, LSTM is able to regulate the flow of information across time steps, selectively retaining or discarding past knowledge depending on the relevance of new observations. This unique capability makes LSTM especially effective in sequential learning tasks, including network traffic analysis, where meaningful patterns often span both short- and long-range temporal contexts. For anomaly detection in IoT and CPS, where attack signatures may be hidden in subtle temporal variations, LSTM provides a powerful foundation for distinguishing between normal and malicious activities. The general architecture of an LSTM layer is illustrated in Figure 1, where an input sequence is processed step by step through memory blocks. At each time step, the network updates its hidden state and cell state, which are propagated forward to influence subsequent computations. The feedback loop ensures that historical context is preserved while new input features are incorporated, allowing the network to maintain a dynamic balance between remembering and forgetting information [39].

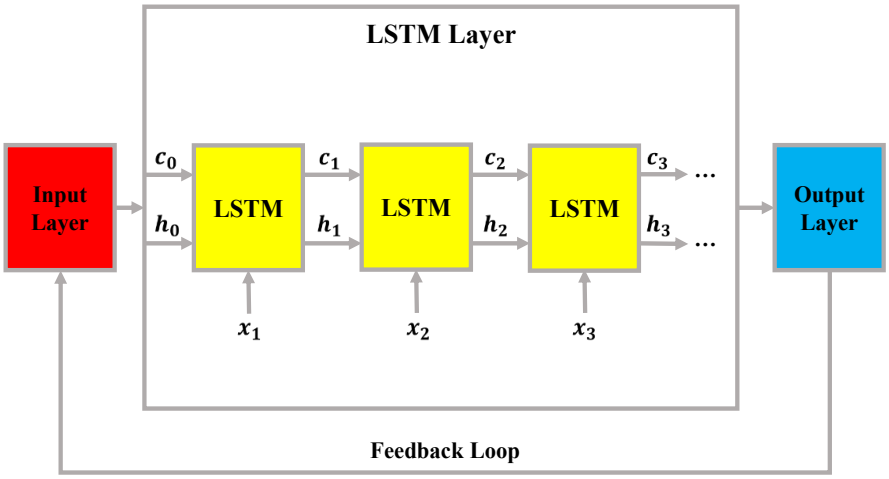

Figure 1: General architecture of the LSTM layer

The internal dynamics of an LSTM unit are regulated by a set of gates that manage the flow of information through the cell. Specifically, the forget gate, formulated in Equation (1), determines which parts of the previous memory should be discarded or retained to maintain relevant long-term information [39]:

$$f_t = \sigma\left(W_{hf}h_{t-1} + W_{if}x_t + B_{hf} + B_{if}\right), \quad (1)$$

Where f_t denotes the forget gate; h_t is updated hidden state; W_{hf}, W_{if}, are weight matrices; B_{hf}, B_{if} are biases terms. Next, the candidate state, expressed in Equation (2), generates a new set of potential memory values using a non-linear transformation of the current input and the previous hidden state [39]:

$$g_t = tanh\left(W_{hg}h_{t-1} + W_{ig}x_t + B_{hg} + B_{ig}\right), \quad (2)$$

Where g_t denotes the candidate generation W_{hg}, W_{ig} are weight matrices; B_{hg}, B_{ig} are biases terms. The input gate, shown in Equation (3), regulates how much of this candidate state should be added to the memory cell:

$$i_t = \sigma(W_{hi}h_{t-1} + W_{ii}x_t + B_{hi} + B_{ii}) \quad (3)$$

 doi:10.20944/preprints202512.0763.v1

Where i_t represents the input gate; W_{hi}, W_{ii} are weight matrices; B_{hi}, B_{ii} are biases terms. Similarly, the output gate, defined in Equation (4), controls which parts of the updated cell state will be exposed as the hidden state for the current time step:

$$o_t = \sigma(W_{ho}h_{t-1} + W_{io}x_t + B_{ho} + B_{io}), \quad (4)$$

Where o_t represents the output gate; W_{ho}, W_{io} are weight matrices; B_{ho}, B_{io} are biases terms. The modulated candidate state, obtained by the interaction between the input gate and the candidate state, is computed as in Equation (5):

$$g'_t = g_t \odot i_t, \quad (5)$$

Where g'_t is modulated candidate state. The cell state is then updated by combining the retained information from the previous memory with the modulated candidate, as given in Equation (6):

$$c_t = (f_t \odot c_{t-1}) + g'_t, \quad (6)$$

Where c_t is updated cell state. Finally, the hidden state, which serves as the short-term representation passed to subsequent layers, is derived according to Equation (7):

$$h_t = o_t \odot tanh(c_t) \quad (7)$$

Where h_t is updated hidden state.

Through this sequential gating mechanism, the LSTM cell adaptively balances long-term memory retention with short-term updates, enabling robust modeling of complex temporal dependencies in sequential data. A more detailed view of the internal mechanism of the LSTM cell is depicted in Figure 2. The diagram highlights the flow of input features, previous hidden state, and previous cell state through the various gates and activation functions. Each gate applies nonlinear transformations to regulate information flow, while the cell state acts as a persistent memory line updated at each step. The interplay of these gates ensures that the model can capture both local fluctuations and long-term dependencies, making LSTM particularly suitable for anomaly detection in sequential IoT data streams where temporal precision is critical [39].

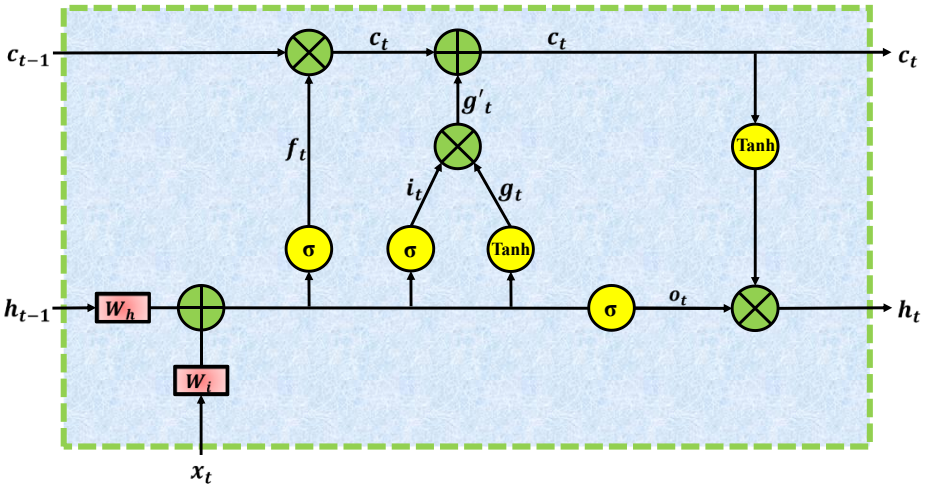

Figure 2: Internal structure and gating mechanism of an LSTM cell

2.3. Transformer Encoders

The Transformer encoder was originally proposed by Vaswani et al. in 2017 as a novel sequence modeling architecture. Unlike earlier recurrent or convolutional approaches, it is entirely based on attention mechanisms, enabling the model to directly capture global dependencies across input sequences without relying on recurrence or local receptive fields. This design provides a significant

 doi:10.20944/preprints202512.0763.v1

advantage in terms of parallelization and computational efficiency, making it highly scalable for large datasets and long input sequences [40]. For anomaly detection in IoT and CPS environments, the Transformer encoder is particularly suitable because it can capture both local and long-range contextual patterns in traffic data. Attack behaviors often manifest across dispersed positions in a sequence, and the self-attention mechanism allows the model to dynamically focus on the most relevant components, improving robustness against stealthy or evolving threats. The operation of the Transformer encoder begins with positional encoding, which provides the model with information about the order of sequence elements. The encoding of even and odd indices is defined in Equations (8) and (9), where sinusoidal functions generate continuous, non-learned embeddings that enable the model to infer sequence position [40]:

$$PE_{(pos,2i)} = sin\left(\frac{pos}{1000^{2i/d}}\right), \tag{8}$$

$$PE_{(pos,2i+1)} = cos\left(\frac{pos}{1000^{2i/d}}\right) \tag{9}$$

Here, pos is the position index, i is the dimension index, and d is the embedding size.

Once positional information is integrated, the model computes the query, key, and value matrices through linear projections of the input, as shown in Equations (10)–(12). These representations are central to the attention mechanism, enabling the model to measure relationships between tokens across the sequence:

$$Q = ZW^Q, \tag{10}$$

$$K = ZW^K, \tag{11}$$

$$V = ZW^V \tag{12}$$

Where W^Q, W^K, W^V are learned projection weights, and Z is the input from the previous encoder layer.

The interaction of queries and keys is then used to calculate the scaled dot-product attention, which measures relevance between elements and normalizes the scores to produce attention weights. This process is described in Equation (13) [40]:

$$Attention\ (Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_K}}\right)V \tag{13}$$

Here, d_K is the dimensionality of each attention head.

To enhance representational power, multiple attention heads are computed in parallel. Their outputs are concatenated and linearly transformed, as expressed in Equation (14). Each head is itself an instance of the attention function defined earlier, as shown in Equation (15):

$$MultiHead\ (Q,K,V) = Concat(head_1, head_2, \dots, head_h)W^O, \tag{14}$$

$$head_1 = Attention\left(QW_i^Q, KW_i^K, VW_i^V\right) \tag{15}$$

Following the attention mechanism, the output is passed through a feed-forward neural network (FFNN) with non-linear activation, as described in Equation (16):

$$FFNN\ (x) = \mathrm{ReLU}(0, xW_1, + b_1)W_2 + b_2 \tag{16}$$

Where x is the input vector corresponding to a single token or position in the sequence, W is the weight matrix, and b is bias vector.

Finally, residual connections and layer normalization are applied to stabilize training and maintain information flow, as formulated in Equations (17) and (18):

 doi:10.20944/preprints202512.0763.v1

$$\acute{Z} = LayerNorm\big(Z + MultiHead\ (Q,K,V)\big), \tag{17}$$

$$Z^{out} = LayerNorm\left(\acute{Z} + FFNN\ \left(\acute{Z}\right)\right) \tag{18}$$

Figure 3 provides a structural overview of the Transformer encoder architecture. It highlights how input embeddings are first enriched with positional encoding, then processed through layers of multi-head self-attention and feed-forward networks, each wrapped with residual connections and normalization. This design allows the model to flexibly capture hierarchical sequence dependencies while ensuring efficient training convergence, making it highly effective for sequential anomaly detection in IoT and CPS traffic data [40].

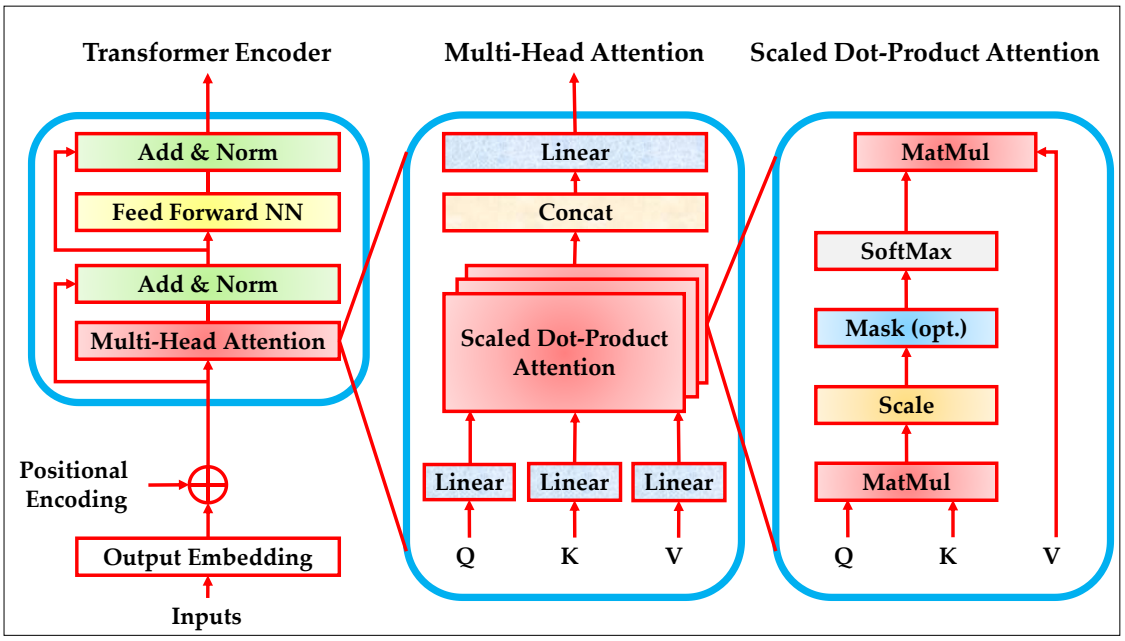

Figure 3: The architecture of the transformer encoder

2.4. IChOA

The ChOA was introduced in 2020 as a bio-inspired meta-heuristic that mimics the intelligent foraging, attacking, and social behaviors of chimpanzees in nature. Unlike swarm-based optimizers, ChOA explicitly models role diversification among agents, making it highly effective in balancing exploration and exploitation. In ChOA, the population is divided into four main categories of chimps: attackers, barriers, chasers, and drivers. Each role represents a different strategy in hunting prey and contributes distinctively to the position-updating mechanism. By imitating this cooperative hunting behavior, ChOA is able to explore the search space broadly while also converging effectively toward optimal solutions [41]. Attackers focus on approaching the prey directly, barriers restrict escape routes, chasers follow the prey's movement, and drivers control the prey's direction. This diverse role assignment enables the algorithm to avoid premature convergence and enhances its global search capability. At every iteration, chimps update their positions relative to the prey, guided by stochastic coefficients that introduce controlled randomness and chaotic adaptation. This design allows ChOA to dynamically balance convergence speed with solution diversity.

Mathematically, the process of encircling the prey and updating positions is captured through Equations (19)–(23). These equations define the distance vector between the prey and the chimp, the iterative update of the chimp's position, and the dynamic coefficients controlling exploration and exploitation. Specifically, Equation (19) represents the distance calculation between prey and chimp, while Equation (20) updates the chimp's position accordingly. Equations (21) and (22) define the adaptive parameters, where random values regulate exploration intensity. Finally, Equation (23)

 doi:10.20944/preprints202512.0763.v1

incorporates chaotic dynamics through m, which prevents stagnation and improves search diversity [41].

$$d = \left|c.X_{prey}(t) - m.X_{chimp}(t)\right| \tag{19}$$

$$X_{chimp}(t+1) = X_{\boldsymbol{prey}}(t) - a.d \tag{20}$$

$$a = 2.f.r_1 - f \tag{21}$$

$$c = 2.r_2 \tag{22}$$

$$m = Chaotic\_value \tag{23}$$

Where $X_{chimp}(t)$ denotes the chimp's position vector; $X_{prey}(t)$ is the prey's position vector; $a, c,$ and m are the coefficient vectors; m indicates a chaotic vector; r_1 and r_2 are the random vectors $\in [0, 1]$; and f is the dynamic vector $\in [0, 2.5]$.

Building upon these foundations, the cooperative behavior of multiple chimps is formulated in Equations (24)–(26). Here, four leaders (attacker, barrier, chaser, and driver) are selected to represent the best candidate solutions. Each role calculates its respective distance vector to the prey as shown in Equation (24). Based on these distances, candidate positions for each role are computed as in Equation (25). The final updated position of a chimp is then derived as the average of these four candidate positions, expressed in Equation (26). This equation ensures that the search process integrates multiple perspectives simultaneously, enhancing convergence accuracy and robustness. The position update mechanism is further visualized in Figure 4, which depicts the coordinated roles of attackers, barriers, chasers, and drivers in guiding the search process. Each role contributes to the encirclement of the prey, thereby simulating the collaborative hunting strategy of chimpanzees [41].

$$\begin{cases} d_{Attacker} = |c_1.X_{Attacker} - m_1.X| \\ d_{Barrier} = |c_2.X_{Barrier} - m_2.X| \\ d_{Chaser} = |c_3.X_{Chaser} - m_3.X| \\ d_{Driver} = |c_4.X_{Driver} - m_4.X| \end{cases} \tag{24}$$

$$\begin{cases} X_1 = X_{Attacker} - a_1(d_{Attacker}) \\ X_2 = X_{Barrier} - a_2(d_{Barrier}) \\ X_3 = X_{Chaser} - a_3(d_{Chaser}) \\ X_4 = X_{Driver} - a_4(d_{Driver}) \end{cases} \tag{25}$$

$$X(t+1) = \frac{X_1 + X_2 + X_3 + X_4}{4} \tag{26}$$

where $X_{Attacker}$ denotes the best search agent, $X_{Barrier}$ presents the second-best search agent, X_{Chaser} presents the third-best search agent, X_{Driver} denotes the fourth-best search agent, and $X(t+1)$ is the updated position of each chimp.

 doi:10.20944/preprints202512.0763.v1

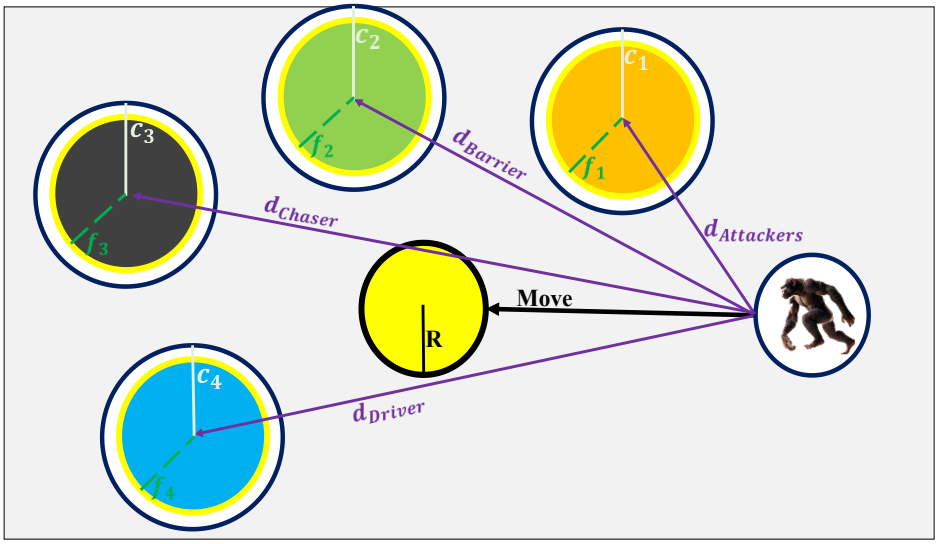

Figure 4: Cooperative hunting mechanism of ChOA

To enhance adaptability, ChOA also employs a modified distance function, as given in Equation (27). This adaptive mechanism allows the algorithm to switch between exploration and exploitation phases depending on the value of chaotic and random coefficients. Figure 5 illustrates this concept, showing how the four roles dynamically adjust their search directions to maintain diversity and avoid local optima. The impact of parameter $|a|$ on convergence and divergence is depicted in Figure 5. When $|a| < 1$, chimps converge toward the prey, intensifying the exploitation phase. Conversely, when $|a| > 1$, the chimps diverge away from the prey, promoting exploration of the wider search space. This dynamic adaptation ensures that ChOA maintains an effective balance between global exploration and local refinement [41].

$$d = \left| c.X_{prey}\,(t) - m.X_{chimp}\,(t) \right| \tag{27}$$

Where μ is the random number ∈ [0, 1].

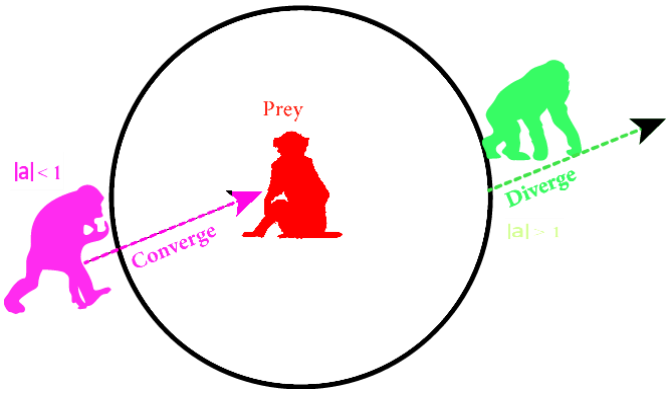

Figure 5: Position updating of chimps and the effect of $|a|$ on convergence and divergence

Although the standard ChOA algorithm has demonstrated strong capabilities in balancing exploration and exploitation, it still exhibits certain limitations. In particular, the reliance on four core roles (attacker, barrier, chaser, and driver) often results in premature convergence, especially when the population tends to cluster around suboptimal regions. Furthermore, while the algorithm excels at exploration in the early iterations, its exploitation strength in the later stages may be insufficient to fine-tune solutions with the required precision. These weaknesses reduce the overall convergence accuracy and stability of the optimization process. To address these shortcomings, an additional chimp role, termed the Refiner, is introduced in the improved version of the algorithm. The refiner represents a highly specialized agent dedicated to intensifying exploitation. Unlike the other roles,

 doi:10.20944/preprints202512.0763.v1

which are distributed between exploration and exploitation, the refiner focuses on refining candidate solutions by operating in closer proximity to the prey. Its task is to ensure that promising regions of the search space are exploited more thoroughly, thereby improving convergence precision.

The inclusion of the refiner also enhances the cooperative dynamics of the chimp group. While the attacker, barrier, chaser, and driver maintain their balance between exploration and exploitation, the refiner acts as a stabilizing force that prevents the swarm from diverging too far during later iterations. By interacting with the other four roles, it reinforces exploitation whenever the algorithm shows signs of stagnation, ensuring that the algorithm converges not only quickly but also accurately. In essence, the refiner functions as an exploitation intensifier, complementing the exploratory behaviors of the other chimps. The updated mathematical formulation of the improved ChOA with the refiner is expressed in Equations (28)–(30). Equation (28) extends the distance calculation by introducing d_{Refiner}, which measures the proximity of the refiner chimp to the prey. Equation (29) incorporates the candidate position, derived from the refiner's update rule. Finally, Equation (30) redefines the overall position update of the chimps as the average of five roles instead of four. This extension ensures that the refiner's influence is explicitly integrated into the position update mechanism, providing additional exploitation pressure without sacrificing the exploratory capacity of the standard ChOA. Through this modification, the improved algorithm strengthens exploitation capabilities while maintaining the diverse exploration offered by the original four chimp roles.

$$\begin{cases} d_{Attacker} = |c_1.X_{Attacker} - m_1.X| \\ d_{Barrier} = |c_2.X_{Barrier} - m_2.X| \\ d_{Chaser} = |c_3.X_{Chaser} - m_3.X| \\ d_{Driver} = |c_4.X_{Driver} - m_4.X| \\ d_{\text{Refiner}} = |c_5.X_{\text{Refiner}} - m_5.X| \end{cases} \tag{28}$$

$$\begin{cases} X_1 = X_{Attacker} - a_1(d_{Attacker}) \\ X_2 = X_{Barrier} - a_2(d_{Barrier}) \\ X_3 = X_{Chaser} - a_3(d_{Chaser}) \\ X_4 = X_{Driver} - a_4(d_{Driver}) \\ X_5 = X_{\text{Refiner}} - a_5(d_{\text{Refiner}}) \end{cases} \tag{29}$$

$$X\,(t+1) = \frac{X_1 + X_2 + X_3 + X_4 + X_5}{5} \tag{30}$$

2.5. Evo-Transformer-LSTM

The proposed Evo-Transformer-LSTM framework integrates the complementary strengths of Transformer encoders, LSTM networks, and the IChOA to construct a unified architecture for anomaly detection in IoT and CPS environments. The overall design, illustrated in Figure 6, begins with raw traffic sequences from the input datasets, which are first preprocessed and then passed through the model pipeline. Each component of the architecture contributes distinctively to capturing the complex temporal and contextual dependencies of network traffic, while the IChOA ensures robust hyper-parameter selection for optimal performance. At the entry point of the architecture, input traffic data sequences are mapped into high-dimensional representations. These embeddings serve as the foundation for capturing meaningful correlations between features. Since IoT and CPS data often exhibit sequential structures with both long-term and short-term dependencies, the model leverages a dual learning strategy that exploits both Transformer-based attention and LSTM memory mechanisms. This integration allows the model to simultaneously focus on global contextual patterns and localized temporal dynamics.

 doi:10.20944/preprints202512.0763.v1

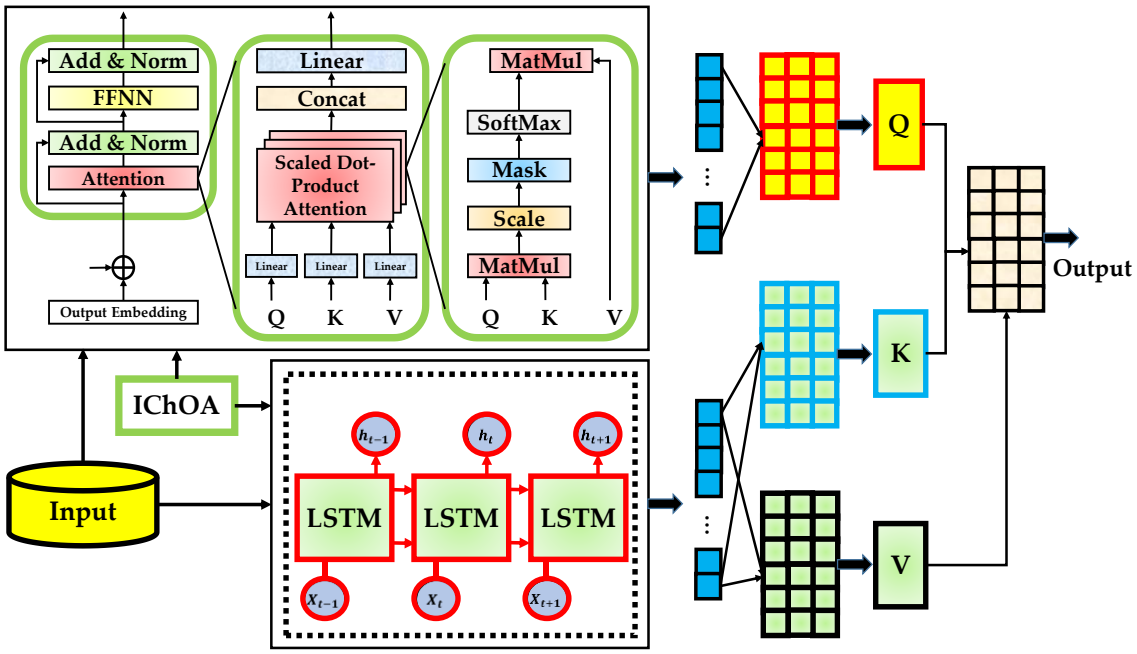

Figure 6: The overall architecture of the proposed Evo-Transformer-LSTM model

The Transformer encoder component plays a critical role in extracting long-range dependencies across the entire traffic sequence. By applying multi-head self-attention, the encoder learns to assign varying levels of importance to different tokens in the sequence, enabling the detection of subtle anomalies that may be dispersed across time. Positional encodings are injected into the input embeddings to preserve sequence order, after which the inputs are linearly projected into query, key, and value matrices. These representations interact through scaled dot-product attention, ensuring that the model highlights the most informative dependencies across features. Following the attention stage, the Transformer applies feed-forward sublayers and residual connections to refine the contextual embeddings. This design provides hierarchical representations that capture not only direct correlations but also higher-level abstractions of the traffic patterns. For anomaly detection, this means the system can recognize both simple deviations and complex attack signatures, even when the anomalies are masked by normal activity. The Transformer output thus encodes global contextual knowledge that forms the foundation for subsequent temporal modeling.

The enriched contextual features are then passed to the LSTM layer, which models sequential dynamics at a finer granularity. While the Transformer excels at learning global relationships, the LSTM complements it by capturing local temporal dependencies and sequential continuity. At each time step, the LSTM processes the embedding along with its hidden and cell states, selectively retaining or discarding information via its gating mechanism. This enables the network to learn how anomalies evolve over time, such as identifying gradual deviations or burst-like abnormal events that require memory of preceding states. The feedback loop within the LSTM ensures that both historical context and new inputs contribute to updated hidden states. This temporal modeling is particularly important in IoT and CPS data, where the difference between normal and abnormal traffic may lie in how patterns unfold over multiple steps. By combining Transformer outputs with LSTM processing, the architecture achieves a balance between capturing broad contextual dependencies and preserving sequential integrity.

To further enhance robustness and generalization, the framework integrates the IChOA as an external optimizer. IChOA tunes critical hyper-parameters of both the Transformer and the LSTM, such as the number of attention heads, embedding dimension, learning rate, and hidden state size. By modeling the cooperative hunting strategies of chimps with adaptive chaotic behavior, IChOA avoids premature convergence and identifies hyperparameter configurations that maximize classification performance. The role of IChOA extends beyond simple tuning; it dynamically balances exploration and exploitation during the optimization process, ensuring that the final Evo-

 doi:10.20944/preprints202512.0763.v1

Transformer-LSTM model is not overfitted to specific datasets but remains adaptable across varying IoT and CPS benchmarks. This evolutionary optimization provides significant performance gains in terms of accuracy, stability, and computational efficiency, as validated in our experimental results.

3. Results

All experiments were implemented in Python 3.10.13, which provides stable compatibility with modern DL libraries. The DL modules, including LSTM and Transformer encoders, were developed using PyTorch 2.3.1 with CUDA 12.1 support, ensuring efficient training on GPU hardware. Supporting libraries included scikit-learn 1.5.0 for preprocessing, evaluation metrics, and baseline comparisons, as well as NumPy 1.26.4 and Pandas 2.2.2 for numerical computation and data manipulation. Visualization and result analysis were performed using Matplotlib 3.9.0 and Seaborn 0.13.2, which allowed consistent graphical representation of performance metrics. The simulations were conducted on a workstation equipped with an NVIDIA RTX 4090 GPU (24 GB VRAM), an Intel Core i9-13900K processor (3.0 GHz, 24 cores), and 128 GB of DDR5 RAM, running on Ubuntu 22.04 LTS. This configuration provided sufficient computational resources to train deep neural models on large-scale datasets such as CIC-IDS-2017 and CSE-CIC-IDS-2018, while also enabling efficient experimentation with the more recent IoT and IoV benchmarks. The hardware environment ensured that both training and inference were executed with reasonable efficiency, supporting extensive hyper-parameter tuning with IChOA without compromising scalability.

To rigorously evaluate the proposed Evo-Transformer-LSTM framework, we compared it against a carefully selected set of widely used ML and DL models. The choice of baselines was motivated by their popularity in anomaly detection research, their ability to represent different families of learning approaches, and their complementary modeling strengths. Together, these baselines form a comprehensive benchmark that allows us to assess whether the proposed hybrid method consistently outperforms both classical and modern approaches to anomaly detection in IoT and CPS environments. From the family of sequential deep models, we included LSTM and Transformer as standalone baselines. LSTM networks, with their gating mechanism, are well-established for capturing temporal dependencies in sequential data, while Transformer encoders, through self-attention, excel at modeling long-range contextual relationships. Importantly, evaluating these architectures individually allows us to investigate how much each contributes on its own, and to quantify the improvement achieved when they are combined in the proposed Evo-Transformer-LSTM framework. In addition, CNN was selected as a baseline since it provides strong capabilities in local feature extraction and pattern recognition, offering a contrasting non-sequential DL perspective.

From the category of advanced representation learning, bidirectional encoder representations from transformers (BERT) was chosen as a baseline. As a bidirectional Transformer model pre-trained on large corpora, BERT has shown strong adaptability in tasks beyond natural language processing, including cybersecurity. Its contextual embedding capabilities provide a powerful comparison point for our proposed architecture. Furthermore, deep reinforcement learning (DRL) was included to represent adaptive policy-based models that optimize decision-making through interaction with the environment. This allows us to benchmark our supervised architecture against adaptive approaches that can learn dynamic anomaly detection strategies. Finally, to represent traditional ML methods, RF and k-nearest neighbors (KNN) were adopted. RF, as an ensemble of decision trees, is widely recognized for its robustness, interpretability, and strong performance in classification tasks, particularly on imbalanced datasets. KNN, though simple, remains an effective distance-based classifier and serves as a lightweight baseline for anomaly detection. Together, RF and KNN ensure that the comparison is not limited to deep models but also covers conventional ML.

For evaluation, we employed a comprehensive set of metrics including accuracy (Acc), F1-score, area under the curve (AUC), root mean squared error (RMSE), runtime performance, variance analysis, and statistical significance testing using the t-test. Together, these metrics capture both quantitative classification performance and qualitative aspects such as convergence stability, computational efficiency, and reliability. Accuracy, defined in Equation (31), measures the proportion

 doi:10.20944/preprints202512.0763.v1

of correctly classified samples relative to the total number of observations. It provides a general indicator of overall correctness by combining both true positives and true negatives. While useful for summarizing overall performance, accuracy can be biased in imbalanced datasets, which makes it essential to interpret it alongside other metrics that are more sensitive to class distribution.

$$Accuracy = \frac{true\ positive + true\ negative}{true\ positive + true\ negative + false\ positive + false\ negative} \quad (31)$$

F1-score, shown in Equation (32), balances precision and recall through their harmonic mean. Precision reflects the ability to minimize false alarms, while recall captures the ability to identify actual attacks. This metric is particularly important in IoT and CPS environments, where an excess of false positives can overwhelm administrators while false negatives may leave attacks undetected. A high F1-score indicates that the model achieves a strong trade-off between detecting anomalies and avoiding spurious alerts.

$$F1 - Score\ = 2 \times \frac{Precision\ \times\ Recall}{Precision\ +\ Recall} \quad (32)$$

AUC, expressed in Equation (33), evaluates the discriminative capacity of the model by integrating the receiver operator characteristic (ROC) curve across all possible thresholds. This threshold-independent measure reflects how consistently the model ranks anomalous traffic above benign traffic. In practice, a higher AUC value suggests greater reliability of the classifier under different operating conditions, which is critical for deployment in diverse IoT and CPS systems where detection thresholds may vary depending on real-time requirements.

$$AUC = \int_0^1 ROC(t)dt \quad (33)$$

Where, $ROC(t)$ is ROC curve at threshold t.

RMSE, defined in Equation (34), quantifies the squared deviation between predicted and actual values. In this study, it was used primarily to track the convergence behavior of the models during training. Lower RMSE values indicate more stable and efficient error reduction, allowing us to compare not only the final classification outcomes but also the learning dynamics. This measure is therefore essential to demonstrate that the proposed Evo-Transformer-LSTM not only outperforms in accuracy but also converges smoothly and reliably.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}[x_i - \hat{x}_i]^2} \quad (34)$$

Where, x_i is the actual value, $\hat{x}_i$ is the predicted value, and N is the total number of data points.

Runtime and variance analysis complement the classification metrics by providing insight into computational efficiency and model stability. Runtime measures the total training and inference time, which is a crucial factor for real-world IoT and CPS deployments where scalability and response speed matter. Variance quantifies the stability of results across multiple runs; low variance indicates that the model produces consistent outcomes, while high variance may reveal sensitivity to initialization or dataset splits. These measures ensure that performance gains are not achieved at the expense of efficiency or reliability. The t-test analysis was used to validate the statistical significance of the observed improvements. By comparing the performance distributions of the proposed model and baseline methods, the t-test determines whether the differences are likely due to genuine performance superiority rather than random variation. Including this statistical test reinforces the robustness of our conclusions, ensuring that the superiority of Evo-Transformer-LSTM is both empirically demonstrated and statistically validated.

Before training the models, the hyper-parameters of each architecture were carefully tuned to ensure fair and optimized evaluation. Hyper-parameter tuning is a critical step, as the performance of DL models is highly sensitive to settings such as learning rate, batch size, dropout, and optimizer

 doi:10.20944/preprints202512.0763.v1

choice. Poorly chosen values can lead to overfitting, underfitting, slow convergence, or unstable training, while well-optimized hyper-parameters can significantly improve both accuracy and generalization. To achieve reliable optimization, different strategies were applied depending on the architecture. For the proposed Evo-Transformer-LSTM model, hyper-parameters were tuned using the IChOA, which dynamically balances exploration and exploitation to identify near-optimal configurations. For the baseline models (including CNN, BERT, DRL, RF, and KNN) hyper-parameter tuning was performed using a grid search strategy. This ensured systematic exploration of candidate values within predefined ranges, ultimately selecting the settings that yielded the best validation performance.

The optimized hyper-parameter values for each model are summarized in Table 1. For Evo-Transformer-LSTM, the configuration included a learning rate of 0.003, batch size of 32, dropout rate of 0.2, weight decay of 0.02, and GELU activation, with IChOA serving as the optimizer. Transformer-specific parameters such as 8 attention heads, 6 encoder layers, and 2048 hidden feed-forward units were tuned, while LSTM layers used 64 hidden units and a sequence length of 6. In the CNN baseline, optimal performance was achieved with 8 convolution layers, kernel size of 3×3, max pooling of size 2×2, and 32 neurons. For BERT, the optimized setup included a learning rate of 0.001, batch size of 64, dropout rate of 0.2, GELU activation, and Adam optimizer, with 8 self-attention heads and 8 encoder layers. DRL was tuned with a learning rate of 0.004, batch size of 64, sigmoid activation, discount factor $\gamma = 0.96$, and ε-greedy = 0.41. RF performed best with 300 estimators, maximum tree depth of 10, and a minimum of 4 samples per split. Finally, KNN achieved its optimal setting with 6 neighbors, Euclidean distance metric, uniform weights, and the Kd-tree algorithm. These optimized configurations, reported concisely in Table 1, represent the best-performing hyper-parameter sets identified for each algorithm and were consistently used in all subsequent experiments.

Table 2 presents the comparative performance of the proposed Evo-Transformer-LSTM against a set of baseline algorithms across four benchmark datasets. The reported metrics include accuracy, F1-score, and AUC, offering both quantitative insights into classification correctness and qualitative evidence of robustness in detecting anomalies. On the most challenging dataset, CICIoV 2024, Evo-Transformer-LSTM achieved an accuracy of 96.72%, F1-score of 97.18%, and an AUC of 98.52%, significantly outperforming traditional baselines such as RF (Accuracy 81.55%, AUC 83.08) and KNN (Accuracy 80.09%, AUC 82.34). This improvement reflects the framework's ability to handle heterogeneous vehicular traffic patterns where attack signatures are often subtle and temporally dispersed. Qualitatively, the high F1-score indicates that the model can successfully capture even rare attack instances, a critical requirement for IoV systems. For CIC IoT-DIAD 2024, which emphasizes IoT traffic with lightweight protocols and irregular communication, Evo-Transformer-LSTM achieved an accuracy of 97.01%, F1-score of 97.62%, and an AUC of 98.15%. The performance margin over baselines such as CNN (Accuracy 84.45%, AUC 86.68) and LSTM (Accuracy 86.03%, AUC 88.05) demonstrates that the hybrid model can effectively capture both global contextual dependencies and local sequential patterns. The results show that the integration of Transformer and LSTM modules allows the system to overcome challenges posed by IoT-specific communication patterns, which often appear noisy and inconsistent.

Table 1: Parameter setting of proposed algorithms

Model	Parameter	Value
Evo-Transformer-LSTM	Learning rate	0.003
	Batch size	32
	Feed forward hidden size	2048
	Weight decay	0.02
	Dropout rate	0.2
	Number of attention heads	8
	Number of encoder layers	6
	Activation function	GELU
	Optimizer	IChOA

 doi:10.20944/preprints202512.0763.v1

CNN	Sequence length	6
	Hidden units per layer	64
	a	[−1, 1]
	f	Linearly from 2 to 0
	Population size	140
	Iteration	300
	Number of convolution layers	8
	Kernel size	3*3
	Pooling type	Max pooling (2*2)
	Number of neurons	32
	Learning rate	0.001
	Batch size	64
	Dropout rate	0.2
BERT	Number of self-attention heads per layer	8
	Number of transformer encoder layers	8
	Length of input time-series window	64
	Activation Function	GELU
	Optimizer	Adam
	Learning rate	0.004
DRL	Discount factor (γ)	0.96
	ε-greedy	0.41
	Batch size	64
	Activation Function	Sigmoid
RF	Number of estimators	300
	Maximum depth of trees	10
	Minimum samples per split	4
KNN	Number of neighbors	6
	Distance metric	Euclidean distance
	Weights	Uniform
	Algorithm	Kd-tree

On CSE-CIC-IDS 2018, the Evo-Transformer-LSTM reached 97.88% accuracy, 98.30% F1-score, and 98.91% AUC, outperforming BERT (Accuracy 91.80%, AUC 93.77) and Transformer (Accuracy 90.19%, AUC 92.41). This dataset contains a wide variety of modern attack types, including cryptomining and botnet-based threats. The superior performance here illustrates how the evolutionary optimization of hyper-parameters by IChOA contributes to stability and adaptability across diverse attack families. The qualitative takeaway is that the model generalizes well even in environments with complex multi-class attack distributions. For the relatively more established CIC-IDS 2017, the Evo-Transformer-LSTM achieved its highest performance, with 98.25% accuracy, 98.83% F1-score, and 99.36% AUC. While other models such as BERT and Transformer also performed strongly (above 92% accuracy and 92% AUC), the proposed framework maintained a clear margin. This outcome can be attributed to the dataset's more standardized attack patterns, which are easier to learn, thus allowing the hybrid architecture to reach near-optimal detection. The high AUC underscores the system's ability to discriminate effectively across benign and malicious flows, even at varying thresholds.

 doi:10.20944/preprints202512.0763.v1

Table 2: Comparative performance of proposed models across four CIC datasets

Proposed Model	Datasets											
	CICIoV 2024			CIC IoT-DIAD 2024			CSE-CIC-IDS 2018			CIC-IDS 2017		
	Acc	F1	AUC	Acc	F1	AUC	Acc	F1	AUC	Acc	F1	AUC
Evo-Transformer-LSTM	96.72	97.18	98.52	97.01	97.62	98.15	97.88	98.30	98.91	98.25	98.83	99.36
BERT	90.36	91.20	92.33	91.25	92.32	93.49	91.80	92.19	93.77	92.46	93.73	92.99
Transformer	89.48	90.49	91.20	90.29	91.28	92.35	90.19	91.66	92.41	91.52	92.25	93.58
DRL	88.88	89.11	90.76	89.81	90.40	91.77	89.55	90.07	91.25	90.14	91.85	91.55
LSTM	85.96	86.08	86.14	86.03	87.18	88.05	86.63	88.38	89.65	87.79	88.33	89.61
CNN	84.18	85.56	86.83	84.45	85.75	86.68	84.79	85.96	86.09	85.43	86.21	87.27
RF	81.55	82.31	83.08	80.23	81.49	82.28	82.08	83.41	84.60	83.28	84.08	85.83
KNN	80.09	81.70	82.34	81.19	82.61	83.43	81.11	82.30	83.44	82.79	83.18	84.12

Figure 7 visualizes the accuracy results reported in Table 2 using radar plots for the four datasets. This graphical representation highlights the comparative performance of Evo-Transformer-LSTM against other baseline models. The radar shape for Evo-Transformer-LSTM consistently dominates the others, reflecting its superior classification accuracy across all datasets. By displaying the results in this format, it becomes easier to observe the performance margins between models and to identify how closely certain baselines approach the proposed framework. The plots reveal that the margin of improvement is most significant in the more challenging datasets, CICIoV 2024 and CIC IoT-DIAD 2024, where simpler models such as RF and KNN remain closer to the inner rings of the radar space. In contrast, for the less complex datasets, CSE-CIC-IDS 2018 and CIC-IDS 2017, the baseline models also achieve relatively high accuracies, yet Evo-Transformer-LSTM model still retains a clear edge. This visualization confirms that the proposed hybrid architecture maintains consistent superiority regardless of dataset complexity, with the largest advantages manifesting in domains where traffic patterns are irregular and attack signatures are more difficult to distinguish.

Figure 8 illustrates the F1-score and AUC values for all models across the four CIC benchmark datasets, providing a visual complement to the numerical results already presented in Table 2. By displaying these metrics as grouped bar charts, the figure highlights both the relative and absolute margins of improvement achieved by the Evo-Transformer-LSTM compared to baselines. The consistent dominance of the proposed method in both F1-score and AUC is clearly observable, reinforcing its strength in balancing precision and recall while maintaining robust discriminative ability. The analysis of the bar charts shows that the performance gap is most pronounced in the more challenging datasets, CICIoV 2024 and CIC IoT-DIAD 2024. In these cases, classical baselines such as RF and KNN remain closer to the lower end of the scale, while deeper models like BERT and Transformer approach higher ranges but still fall short of Evo-Transformer-LSTM. For CSE-CIC-IDS 2018 and CIC-IDS 2017, although all models achieve relatively strong F1-scores and AUC values, the proposed architecture consistently leads with visible margins, confirming that its hybrid design not only performs well under difficult conditions but also achieves near-optimal results in more structured datasets.

 doi:10.20944/preprints202512.0763.v1

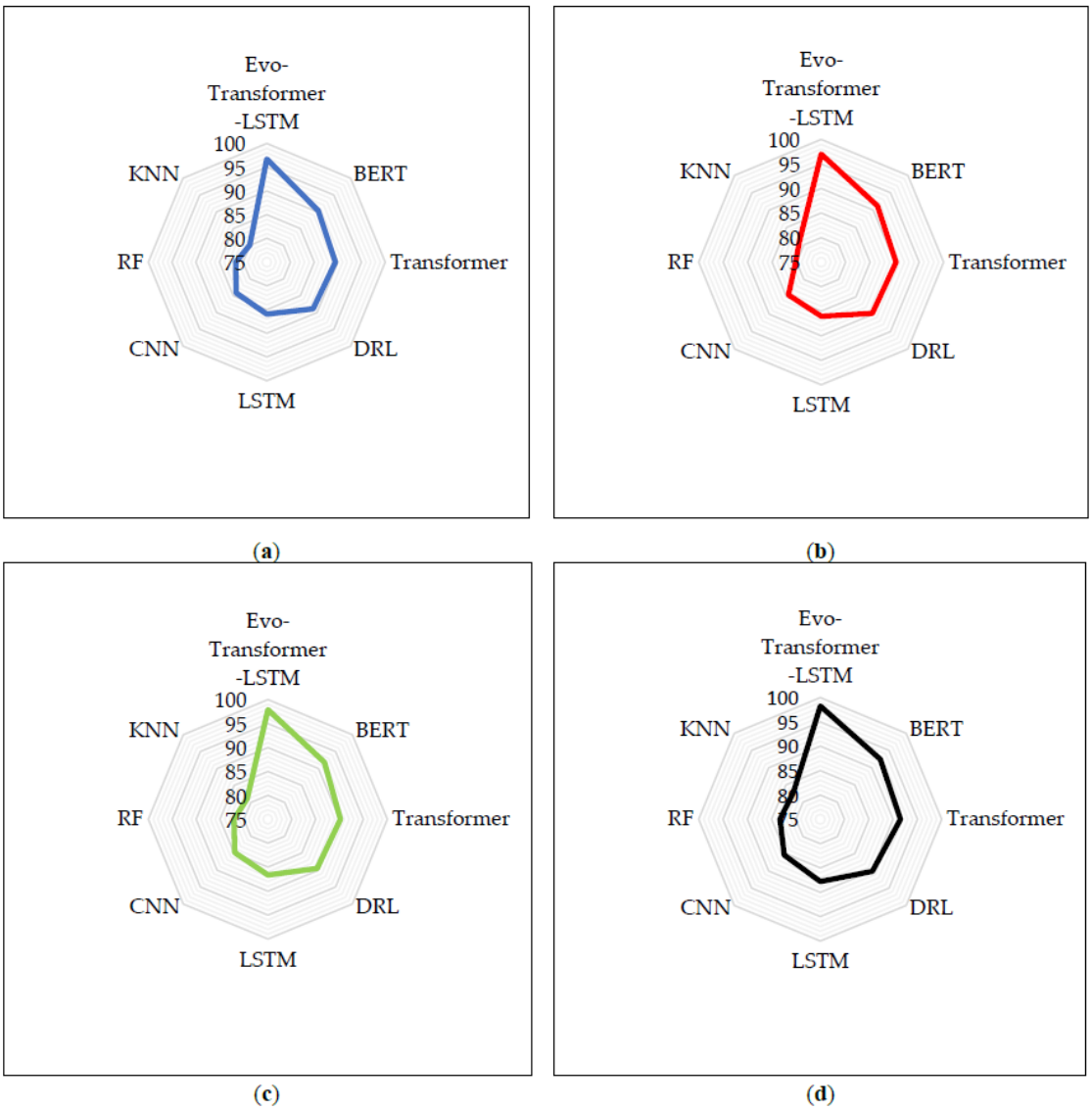

Figure 7: Radar plot comparison of accuracy across models on four CIC datasets: (**a**) CICIoV 2024; (**b**) CIC IoT-DIAD 2024; (**c**) CSE-CIC-IDS 2018; (**d**) CIC-IDS 2017.

Figure 9 presents the ROC curves of the proposed Evo-Transformer-LSTM and baseline models across the four CIC datasets. By plotting sensitivity (true positive rate) against 1-specificity (false positive rate), the figure provides a threshold-independent perspective on classifier performance. The ROC curves allow us to visually compare the discriminative ability of each model, confirming the trends already suggested by the AUC values in Table 2. The ROC curves clearly show that Evo-Transformer-LSTM consistently maintains the steepest ascent and largest enclosed AUC across all datasets. This pattern indicates that the model achieves both high sensitivity and specificity, meaning it effectively detects attacks without generating excessive false alarms. In contrast, traditional baselines such as RF and KNN exhibit much flatter curves, highlighting their limited capacity to balance true positive and false positive rates, especially in the more complex IoV and IoT scenarios. Deep models like BERT and Transformer approach stronger ROC behavior, but their curves still fall short of the sharper separation achieved by the hybrid Evo-Transformer-LSTM model.

 doi:10.20944/preprints202512.0763.v1

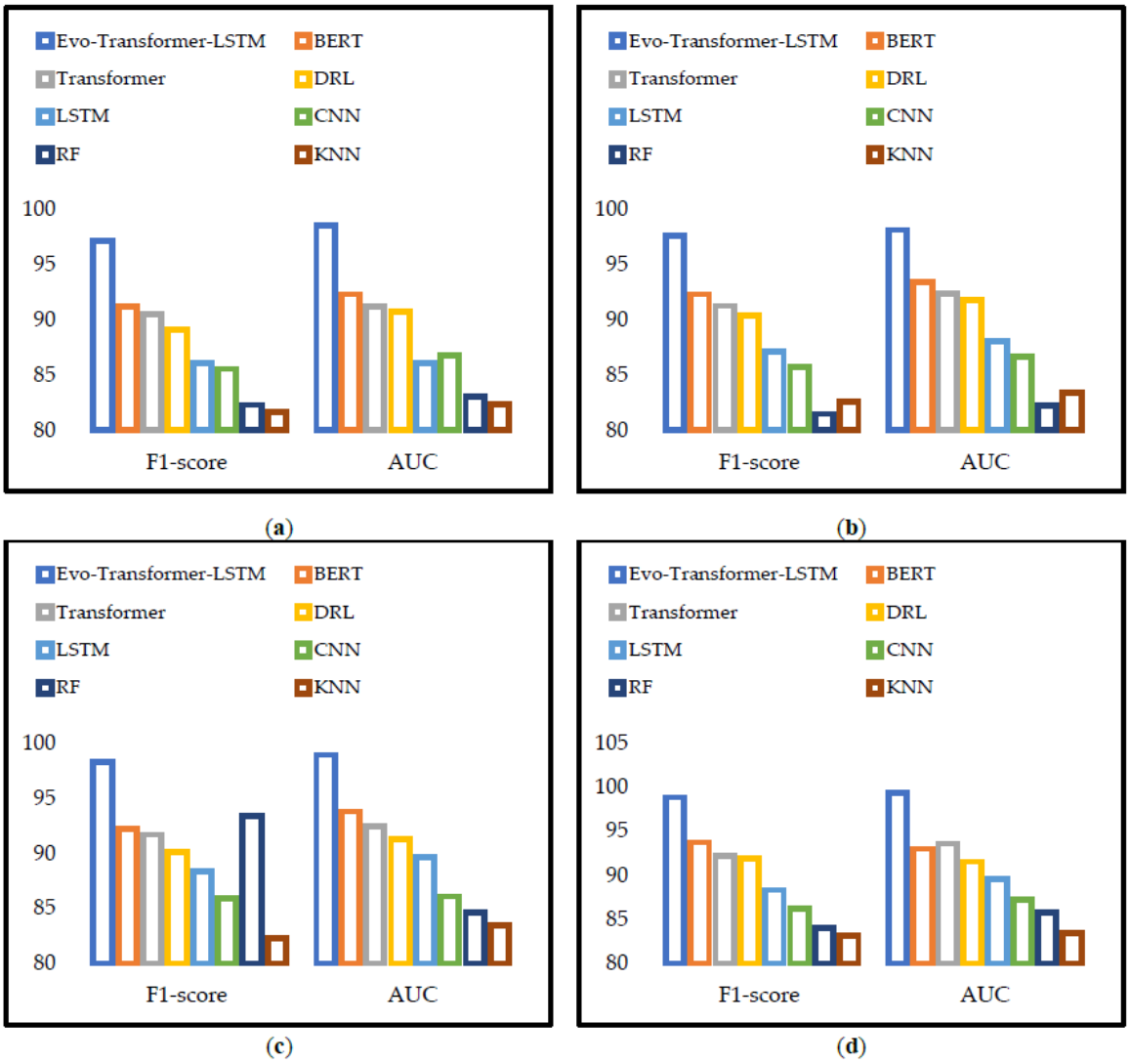

Figure 8: Bar chart visualization of F1-score and AUC across four CIC datasets: (**a**) CICIoV 2024; (**b**) CIC IoT-DIAD 2024; (**c**) CSE-CIC-IDS 2018; (**d**) CIC-IDS 2017.

The superior ROC performance of the proposed model can be directly linked to the synergy of its components. The Transformer encoder provides global contextual awareness, the LSTM model captures sequential temporal dependencies, and IChOA algorithm optimizes hyper-parameters to balance learning stability and generalization. This combination produces a curve that consistently dominates across datasets, reflecting robustness not only in conventional traffic (CIC-IDS 2017, CSE-CIC-IDS 2018) but also in the highly dynamic and irregular domains of IoT-DIAD and CICIoV. These results demonstrate that the hybrid architecture leverages the complementary strengths of its constituent models, enabling it to outperform single-method baselines that lack either contextual depth, temporal continuity, or optimization-driven adaptability.

 doi:10.20944/preprints202512.0763.v1

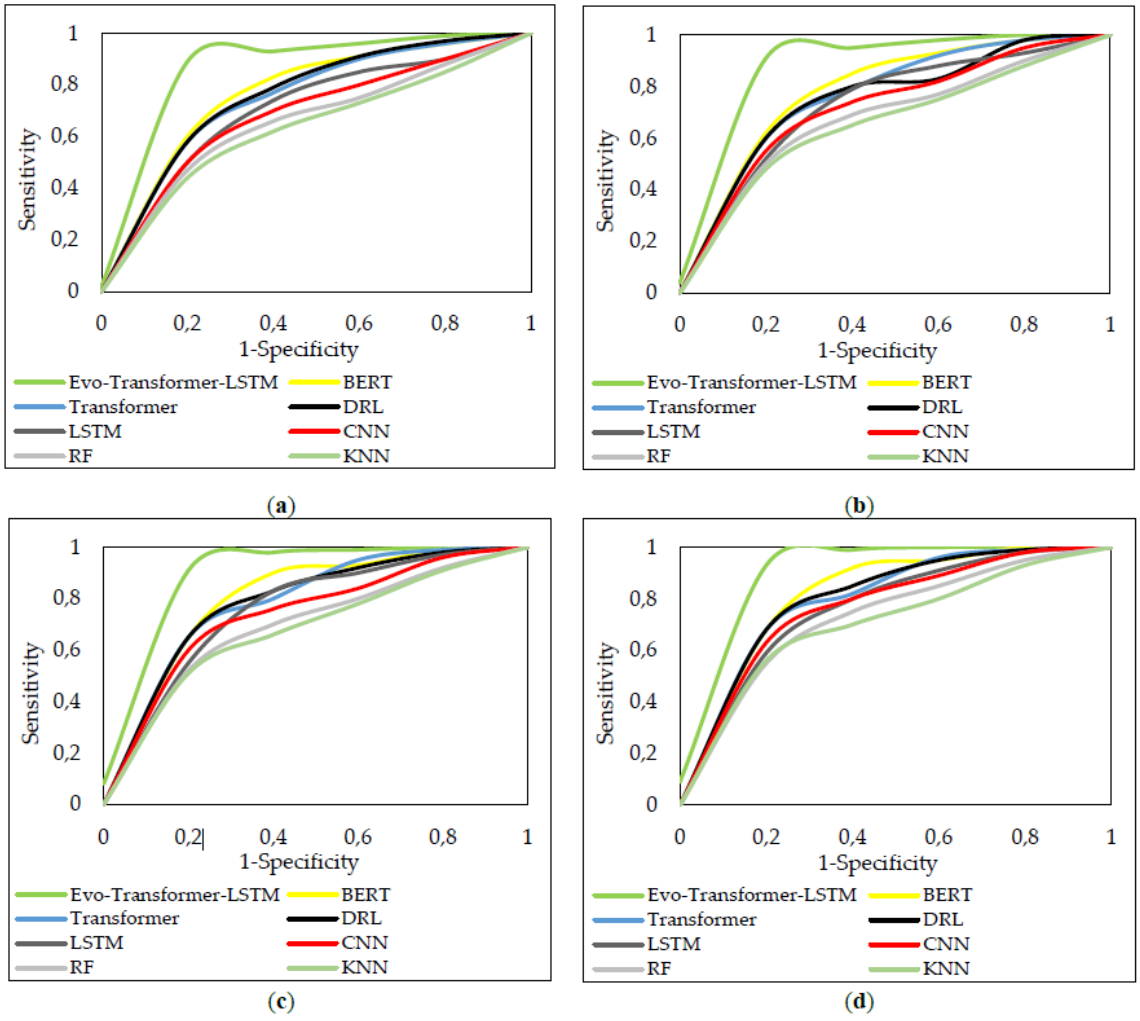

Figure 9: ROC curves of proposed models across four CIC datasets: (**a**) CICIoV 2024; (**b**) CIC IoT-DIAD 2024; (**c**) CSE-CIC-IDS 2018; (**d**) CIC-IDS 2017.

Figure 10 illustrates the training convergence behavior of all models by plotting RMSE against epochs across the four CIC datasets. These curves provide insight into the stability and efficiency of the training process, showing not only the final error levels but also the speed at which each model approaches convergence. The comparison highlights the ability of different architectures to minimize error over time, directly linking their learning dynamics to robustness in classification. The results show that the proposed Evo-Transformer-LSTM achieves the fastest and most stable convergence across all datasets. Its RMSE drops sharply within the first ~50 epochs and stabilizes near zero well before 100 epochs, reflecting both efficient error reduction and strong optimization through IChOA. In contrast, single deep models such as Transformer, LSTM, and CNN converge more slowly, typically requiring 150–250 epochs to reach a plateau, while BERT achieves moderate improvement but still lags behind the hybrid framework. Traditional models such as RF and KNN converge the slowest and exhibit higher residual error, underscoring their limited capacity to capture complex feature interactions.

 doi:10.20944/preprints202512.0763.v1

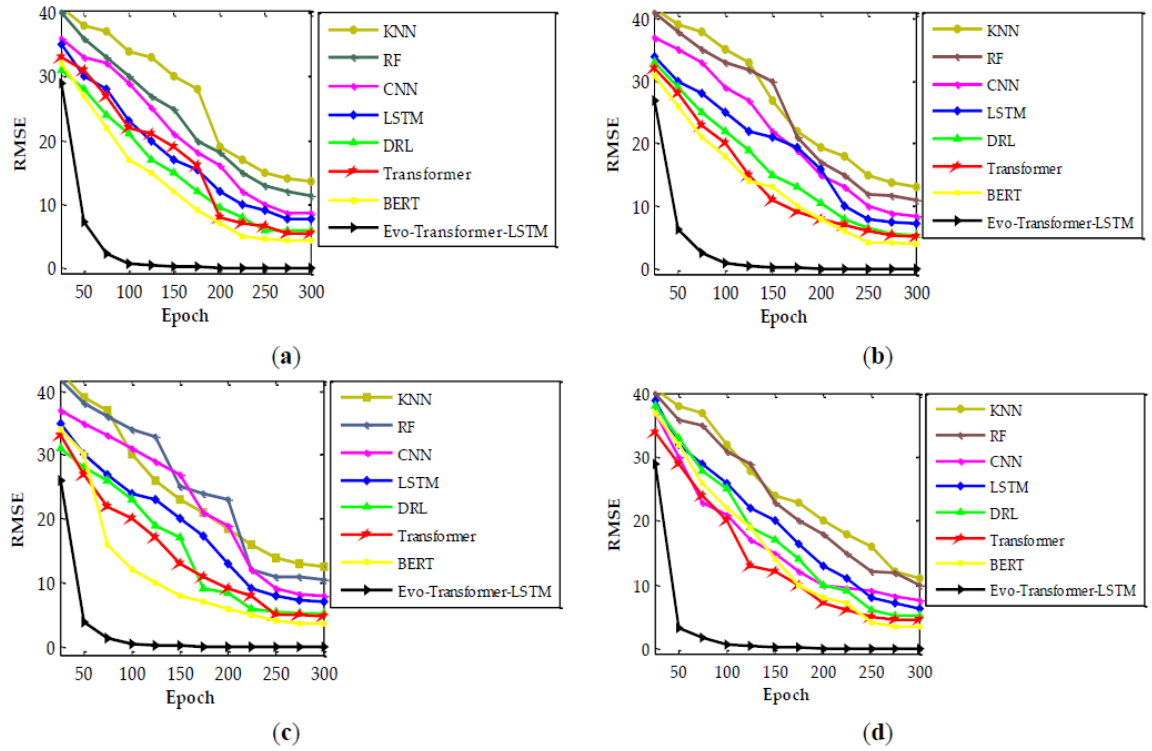

Figure 10: Training convergence curves of RMSE over epochs for the four CIC datasets: (**a**) CICIoV 2024; (**b**) CIC IoT-DIAD 2024; (**c**) CSE-CIC-IDS 2018; (**d**) CIC-IDS 2017.

The proposed model achieved higher accuracy, F1-score, and AUC values compared to both DL and traditional baselines, while also exhibiting faster and more stable convergence in terms of RMSE. These findings confirm that the integration of Transformer encoders, LSTM modules, and evolutionary hyper-parameter tuning provides a robust and scalable solution for anomaly detection in IoT and CPS environments. Building on these results, it is essential to further analyze additional aspects that determine the practicality and generalizability of the framework. To strengthen the reliability of the findings, a statistical significance analysis was performed using the t-test. While raw metrics such as accuracy or AUC demonstrate performance differences, statistical testing ensures that the observed improvements are not the result of random variation but reflect genuine superiority of the proposed model. This evaluation is particularly important in security-sensitive domains, where decision-making must be based on consistently verifiable results rather than single-run outcomes. Beyond statistical validation, model stability and computational efficiency are also critical for real-world deployment. Variance analysis across multiple runs provides insight into how reliably a model maintains its performance under different training conditions, while runtime evaluation highlights whether a model can scale to time-sensitive applications such as IoT intrusion detection or vehicular anomaly detection. Together, these evaluations extend the discussion beyond raw accuracy and F1-score, offering a more comprehensive perspective on whether the proposed Evo-Transformer-LSTM can be generalized and applied effectively in diverse real-world settings.

Table 3 reports the statistical significance analysis using t-tests, comparing the proposed Evo-Transformer-LSTM against all baseline models across the four CIC datasets. The results consistently show extremely low p-values (all below 0.001), which confirms that the performance improvements of the proposed model are statistically significant rather than due to random variation. This validation is critical, as it demonstrates that the superior performance observed in accuracy, F1-score, and AUC metrics is reliably repeatable across independent runs. The analysis also highlights that the margin of statistical significance holds across both DL baselines such as BERT, Transformer, and DRL, as well as traditional ML methods like RF and KNN. This finding underscores the robustness of the Evo-Transformer-LSTM, showing that its improvements are not limited to a specific category of algorithms but extend across fundamentally different modeling paradigms. In practical terms, this

 doi:10.20944/preprints202512.0763.v1

statistical evidence strengthens confidence that the proposed model can consistently outperform existing methods when applied to diverse and complex datasets in IoT and CPS anomaly detection.

Table 3: Results of statistical t-tests comparing the Evo-Transformer-LSTM with baseline models

Model	Statistical t-tests							
	CICIoV 2024		CIC IoT-DIAD 2024		CSE-CIC-IDS 2018		CIC-IDS 2017	
	p-value	Results	p-value	Results	p-value	Results	p-value	Results
Proposed vs. BERT	0.0008	Significant	0.0006	Significant	0.0005	Significant	0.0003	Significant
Proposed vs. Transformer	0.0007	Significant	0.0007	Significant	0.0008	Significant	0.0001	Significant
Proposed vs. DRL	0.0003	Significant	0.0005	Significant	0.0004	Significant	0.0002	Significant
Proposed vs. LSTM	0.00006	Significant	0.00004	Significant	0.00007	Significant	0.00004	Significant
Proposed vs. CNN	0.00005	Significant	0.00002	Significant	0.00006	Significant	0.00007	Significant
Proposed vs. RF	0.000008	Significant	0.000007	Significant	0.000005	Significant	0.000006	Significant
Proposed vs. KNN	0.000006	Significant	0.000005	Significant	0.000002	Significant	0.000003	Significant

Table 4 reports the variance of performance scores for the proposed Evo-Transformer-LSTM and baseline models, computed across 30 independent training runs for each dataset. Models with lower variance are more reliable, as they are less affected by random initialization and stochastic training conditions, while higher variance reflects instability and potential sensitivity to noise. The results demonstrate that Evo-Transformer-LSTM consistently yields the lowest variance across all datasets, with values close to zero (e.g., 0.00083 for CICIoV 2024 and 0.00043 for CIC-IDS 2017). This outcome highlights the robustness of the proposed architecture, showing that it not only achieves superior accuracy and AUC but also maintains highly stable results across repeated runs. By contrast, traditional ML methods such as RF and KNN exhibit the highest variance (above 8.0 in most cases), confirming their limited resilience under varying conditions. DL baselines like BERT, Transformer, and DRL model achieve moderate variance, yet still fall short of the stability offered by the hybrid model. These findings underscore that the proposed framework is not only accurate but also dependable, a crucial advantage for real-world IoT and CPS anomaly detection tasks where consistency across multiple deployments is essential.

Table 4: Variance of model performance across four CIC datasets over 30 independent runs

Method	Variance			
	CICIoV 2024	CIC IoT-DIAD 2024	CSE-CIC-IDS 2018	CIC-IDS 2017
Evo-Transformer-LSTM	0.00083	0.00065	0.00071	0.00043
BERT	1.82548	1.16325	1.65412	2.41523
Transformer	2.16521	2.02456	1.96325	2.02531
DRL	3.95324	3.24532	2.47856	3.14586
LSTM	5.24153	5.01589	6.21458	5.02586
CNN	6.52369	6.32145	5.75632	6.17563
RF	9.81253	8.63521	8.45896	8.53014
KNN	10.32856	9.25478	9.63521	9.75301

Table 5 summarizes the computational efficiency of the proposed Evo-Transformer-LSTM compared with baseline models, measured in terms of run time until different convergence thresholds of RMSE (<20, <15, <10, and <5) were reached. This analysis provides an important perspective beyond predictive performance, as it evaluates the feasibility of deploying these models in real-world IoV scenarios where computational time is a critical factor. The results show that Evo-

 doi:10.20944/preprints202512.0763.v1

Transformer-LSTM is consistently the fastest model to reach all termination thresholds. For instance, it achieves RMSE <20 in only 35 seconds and RMSE <5 in 342 seconds, whereas none of the baseline models are able to converge to the strictest RMSE <5 threshold within practical time limits. By contrast, traditional methods such as RF and KNN exhibit significantly longer times or even fail to converge under tighter thresholds, reflecting their limited efficiency in handling high-dimensional traffic data.

Similarly, DL baselines like BERT, Transformer, and DRL model require substantially more time (over 400–900 seconds for RMSE <15 or <10), making them less suitable for time-sensitive anomaly detection. These findings highlight that Evo-Transformer-LSTM not only ensures superior accuracy and stability but also offers remarkable efficiency in terms of convergence speed. This property is particularly advantageous for IoV environments, where rapid adaptation and real-time decision-making are essential to maintaining security and reliability. The ability to converge faster and under stricter error thresholds reinforces the practicality of the proposed framework for large-scale, dynamic deployments.

Table 5: Comparison of run time for different models on the CICIoV 2024 dataset

Method	Run Time (s)			
	RMSE < 20	**RMSE < 15**	**RMSE < 10**	**RMSE < 5**
Evo-Transformer-LSTM	35	92	163	342
BERT	163	426	793	-
Transformer	152	493	853	-
DRL	109	528	970	-
LSTM	185	763	-	-
CNN	180	805	-	-
RF	205	-	-	-
KNN	271	-	-	-

4. Conclusions

In this study, we introduced a novel anomaly detection framework, Evo-Transformer-LSTM, that unifies the contextual modeling capabilities of Transformer encoders, the sequential memory strength of LSTM, and the hyper-parameter optimization capacity of the IChOA. The architecture was carefully designed to handle the heterogeneous and dynamic traffic characteristics of IoT and CPS environments, and it was validated extensively on four benchmark datasets: CIC-IDS 2017, CSE-CIC-IDS 2018, CIC IoT-DIAD 2024, and CICIoV 2024. By integrating these complementary components, the proposed framework aimed to overcome the weaknesses of individual models, delivering a robust, scalable, and adaptive solution for modern anomaly detection challenges.

The quantitative results confirmed the effectiveness of the proposed method. Evo-Transformer-LSTM consistently achieved the highest scores across all performance metrics, with accuracy levels exceeding 96% on the most challenging datasets and peaking at 98.25% on CIC-IDS 2017. Similarly, F1-scores above 97% and AUC values nearing 99% demonstrated that the model not only detected anomalies reliably but also balanced precision and recall effectively. These improvements were further supported by statistical significance testing, where all pairwise comparisons yielded p-values < 0.001, confirming that the observed gains were not due to random variation. Variance analysis across 30 independent runs showed that the proposed method maintained near-zero fluctuations, far outperforming baselines such as RF and KNN, which suffered from high instability. Beyond accuracy, the Evo-Transformer-LSTM also demonstrated practical efficiency. Convergence analysis revealed that the model reached stable RMSE levels within the first 50 epochs, much faster than competing deep models, and runtime experiments showed that it could achieve strict error thresholds (RMSE < 5) in less than 350 seconds on CICIoV 2024, while no baseline managed the same. This combination of high predictive performance, statistical robustness, stability, and computational

 doi:10.20944/preprints202512.0763.v1

efficiency highlights the framework's readiness for real-world deployment in time-sensitive IoT and vehicular systems.

A central conclusion of this study is that the strength of Evo-Transformer-LSTM does not solely lie in outperforming individual baselines, but in showing how the fusion of complementary learning paradigms fundamentally changes anomaly detection for IoT and CPS. The Transformer captures global contextual signals, the LSTM secures sequential memory, and the IChOA optimizer ensures that both are fine-tuned for stability and efficiency. This tri-layered integration reveals that the weaknesses of each component, when used alone, can be systematically offset by the strengths of the others. Such a design principle—treating anomaly detection not as a single-model problem but as an interplay of diverse modeling strategies—marks a departure from conventional practices and provides a more holistic foundation for addressing the complexity of heterogeneous traffic environments. Equally important is the evidence that the proposed method achieves robustness across drastically different datasets without tailoring the core architecture to a specific domain. The ability to sustain near-optimal performance in enterprise, IoT, and vehicular contexts shows that Evo-Transformer-LSTM is not just a specialized detector but a generalizable methodology. This robustness suggests that anomaly detection should be viewed less as an exercise in tuning algorithms to isolated datasets and more as developing architectures resilient to domain shifts and data irregularities. By demonstrating that one unified framework can handle such variability, the study highlights a methodological shift toward building anomaly detection systems that are inherently adaptive, reliable, and ready to serve as foundational components in securing real-world cyber-physical infrastructures.

For future work, several promising directions can be pursued to further advance anomaly detection in IoT and CPS. On the methodological side, the Evo-Transformer-LSTM framework can be extended by incorporating graph-based learning to capture topological relationships among devices, or by integrating federated and privacy-preserving mechanisms to address data distribution constraints across edge nodes. The optimization process can also be enriched by exploring multi-objective evolutionary algorithms, balancing not only accuracy and convergence but also energy efficiency and latency, which are crucial in resource-constrained environments. From the application perspective, evaluating the framework on large-scale, real-time streaming data and in cross-domain transfer scenarios would provide valuable insights into its adaptability for smart grids, industrial IoT, and vehicular systems. By combining methodological enhancements with deployment-focused considerations, future research can transform the current framework into a truly end-to-end anomaly detection solution for securing next-generation cyber-physical infrastructures.

Acknowledgement: Not applicable.

Funding Statement: The author(s) received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Pardis Sadatian Moghaddam, Mahyar Mahmoudi, Nuria Serrano, Francisco Hernando-Gallego and Diego Martín; methodology, Pardis Sadatian Moghaddam, Mahyar Mahmoudi and Nuria Serrano; software, Pardis Sadatian Moghaddam, Mahyar Mahmoudi and Diego Martín; validation, Pardis Sadatian Moghaddam, Francisco Hernando-Gallego and Diego Martín; formal analysis, Mahyar Mahmoudi and Nuria Serrano; investigation, Pardis Sadatian Moghaddam, Mahyar Mahmoudi, Nuria Serrano, Francisco Hernando-Gallego and Diego Martín; data curation, Pardis Sadatian Moghaddam and Mahyar Mahmoudi; writing—original draft preparation, Pardis Sadatian Moghaddam, Mahyar Mahmoudi, Nuria Serrano, Francisco Hernando-Gallego and Diego Martín; visualization, Nuria Serrano and Francisco Hernando-Gallego; supervision, Diego Martín; project administration, Diego Martín; funding acquisition, Diego Martín. All authors reviewed the results and approved the final version of the manuscript".

Availability of Data and Materials: The data that support the findings of this study are available from the Corresponding Author, [author initials], upon reasonable request.

 doi:10.20944/preprints202512.0763.v1

Ethics Approval: Not applicable.

Conflicts of Interest: The author(s) declare(s) no conflicts of interest to report regarding the present study.

References

1. Makhetha MJ, Markus ED, Abu-Mahfouz AM. Integration of wireless power transfer and low power wide area networks in IoT applications—a review. Sensors Int. 2024;5:100284. doi:10.1016/j.sintl.2024.100284.
2. Almutairi M, Sheldon FT. IoT–cloud integration security: a survey of challenges, solutions, and directions. Electronics. 2025;14:1394. doi:10.3390/electronics14071394.
3. Ferdous J, Islam R, Mahboubi A, Islam MZ. A survey on ML techniques for multi-platform malware detection: securing PC, mobile devices, IoT, and cloud environments. Sensors. 2025;25:1153. doi:10.3390/s25041153.
4. Singh NJ, Hoque N, Singh KR, Bhattacharyya DK. Botnet-based IoT network traffic analysis using deep learning. Secur Privacy. 2024;7:e355. doi:10.1002/spy2.355.
5. Taherdoost H, Le T-V, Slimani K. Cryptographic techniques in artificial intelligence security: a bibliometric review. Cryptography. 2025;9:17. doi:10.3390/cryptography9010017.
6. Toman ZH, Hamel L, Toman SH, Graiet M, Valadares DCG. Formal verification for security and attacks in IoT physical layer. J Reliab Intell Environ. 2024;10:73–91. doi:10.1007/s40860-023-00202-y.
7. Najafi F, Kaveh M, Mosavi MR, Brighente A, Conti M. EPUF: an entropy-derived latency-based DRAM physical unclonable function for lightweight authentication in Internet of Things. IEEE Trans Mob Comput. 2024;24:2422–36. doi:10.1109/TMC.2024.3494612.
8. Cirne A, Sousa PR, Resende JS, Antunes L. Hardware security for Internet of Things identity assurance. IEEE Commun Surv Tutor. 2024;26:1041–79. doi:10.1109/COMST.2024.3355168.
9. Lotfy A, Kaveh M, Martín D, Mosavi MR. An efficient design of Anderson PUF by utilization of the Xilinx primitives in the SLICEM. IEEE Access. 2021;9:23025–34. doi:10.1109/ACCESS.2021.3056291.
10. Agbor BA, Stephen BUA, Asuquo P, Luke UO, Anaga V. Hybrid CNN–BiLSTM–DNN approach for detecting cybersecurity threats in IoT networks. Computers. 2025;14:58. doi:10.3390/computers14020058.
11. Diana L, Dini P, Paolini D. Overview on intrusion detection systems for computers networking security. Computers. 2025;14:87. doi:10.3390/computers14030087.
12. Alsalman D. A comparative study of anomaly detection techniques for IoT security using adaptive machine learning for IoT threats. IEEE Access. 2024;12:14719–30. doi:10.1109/ACCESS.2024.3359033.
13. Black G, Fronczyk K, Arliss W, Allen R. Descriptor: firewall attack detections and extractions (FADE). IEEE Data Descr. 2025;2:163–72. doi:10.1109/IEEEDATA.2025.3572866.
14. Rafique SH, Abdallah A, Musa NS, Murugan T. Machine learning and deep learning techniques for Internet of Things network anomaly detection—current research trends. Sensors. 2024;24:1968. doi:10.3390/s24061968.
15. Inuwa MM, Das R. A comparative analysis of various machine learning methods for anomaly detection in cyber attacks on IoT networks. Internet Things. 2024;26:101162. doi:10.1016/j.iot.2024.101162.
16. Ghurab M, Gaphari G, Alshami F, Alshamy R, Othman S. A detailed analysis of benchmark datasets for network intrusion detection system. Asian J Res Comput Sci. 2021;7:14–33. doi:10.9734/AJRCOS/2021/v7i430185.
17. Bakhshi T, Ghita B. Anomaly detection in encrypted Internet traffic using hybrid deep learning. Secur Commun Netw. 2021;2021:5363750. doi:10.1155/2021/5363750.
18. Najafi Mohsenabad H, Tut MA. Optimizing cybersecurity attack detection in computer networks: a comparative analysis of bio-inspired optimization algorithms using the CSE-CIC-IDS 2018 dataset. Appl Sci. 2024;14:1044. doi:10.3390/app14031044.
19. Yoshimura N, Kuzuno H, Shiraishi Y, Morii M. DOC-IDS: a deep learning-based method for feature extraction and anomaly detection in network traffic. Sensors. 2022;22:4405. doi:10.3390/s22124405.
20. Fathima AN, Ibrahim SS, Khraisat A. Enhancing network traffic anomaly detection: leveraging temporal correlation index in a hybrid framework. IEEE Access. 2024;12:1–15. doi:10.1109/ACCESS.2024.3458903.
21. Saleh SM, Sayem IM, Madhavji N, Steinbacher J. Advancing software security and reliability in cloud platforms through AI-based anomaly detection. In: Proceedings of the 2024 on Cloud Computing Security Workshop; 2024 Nov; New York, NY, USA. p. 43–52. doi:10.1145/3689938.3694779.
22. Engelen G, Rimmer V, Joosen W. Troubleshooting an intrusion detection dataset: the CICIDS2017 case study. In: Proceedings of the 2021 IEEE Security and Privacy Workshops (SPW); 2021 May; San Francisco, CA, USA. p. 7–12. doi:10.1109/SPW53761.2021.00009.
23. Farhan BI, Jasim AD. Performance analysis of intrusion detection for deep learning model based on CSE-CIC-IDS2018 dataset. Indones J Electr Eng Comput Sci. 2022;26:1165–72. doi:10.11591/ijeecs.v26.i2.pp1165-1172.

 doi:10.20944/preprints202512.0763.v1

24. Limon-Cantu D, Alarcon-Aquino V. Multiresolution dendritic cell algorithm for network anomaly detection. PeerJ Comput Sci. 2021;7:e749. doi:10.7717/peerj-cs.749.
25. Selvam R, Velliangiri S. An improving intrusion detection model based on novel CNN technique using recent CIC-IDS datasets. In: Proceedings of the 2024 International Conference on Distributed Computing and Optimization Techniques (ICDCOT); 2024 Mar; Bengaluru, India. p. 1–6. doi:10.1109/ICDCOT61034.2024.10515433.
26. Wang YC, Houng YC, Chen HX, Tseng SM. Network anomaly intrusion detection based on deep learning approach. Sensors. 2023;23:2171. doi:10.3390/s23042171.
27. Khan ZI, Afzal MM, Shamsi KN. A comprehensive study on CIC-IDS2017 dataset for intrusion detection systems. Int Res J Adv Eng Hub. 2024;2:254–60.
28. Lin P, Ye K, Xu CZ. Dynamic network anomaly detection system by using deep learning techniques. In: Proceedings of the International Conference on Cloud Computing; 2019 Jun; Cham, Switzerland. p. 161–76. doi:10.1007/978-3-030-23502-4_12.
29. Yusof MHM, Almohammedi AA, Shepelev V, Ahmed O. Visualizing realistic benchmarked IDS dataset: CIRA-CIC-DoHBrw-2020. IEEE Access. 2022;10:94624–42. doi:10.1109/ACCESS.2022.3204690.
30. Rosay A, Carlier F, Cheval E, Leroux P. From CIC-IDS2017 to LYCOS-IDS2017: a corrected dataset for better performance. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology; 2021 Dec; New York, NY, USA. p. 570–5. doi:10.1145/3486622.3493973.
31. Dehlaghi-Ghadim A, Moghadam MH, Balador A, Hansson H. Anomaly detection dataset for industrial control systems. IEEE Access. 2023;11:107982–96. doi:10.1109/ACCESS.2023.3320928.
32. Saini N, Bhat Kasaragod V, Prakasha K, Das AK. A hybrid ensemble machine learning model for detecting APT attacks based on network behavior anomaly detection. Concurrency Computat Pract Exp. 2023;35:e7865. doi:10.1002/cpe.7865.
33. Mezina A, Burget R, Travieso-González CM. Network anomaly detection with temporal convolutional network and U-Net model. IEEE Access. 2021;9:143608–22. doi:10.1109/ACCESS.2021.3121998.
34. Stiawan D, Idris MYB, Bamhdi AM, Budiarto R. CICIDS-2017 dataset feature analysis with information gain for anomaly detection. IEEE Access. 2020;8:132911–21. doi:10.1109/ACCESS.2020.3009843.
35. Liu L, Engelen G, Lynar T, Essam D, Joosen W. Error prevalence in NIDS datasets: a case study on CIC-IDS-2017 and CSE-CIC-IDS-2018. In: Proceedings of the 2022 IEEE Conference on Communications and Network Security (CNS); 2022 Oct; Austin, TX, USA. p. 254–62. doi:10.1109/CNS56114.2022.9947235.
36. Maseer ZK, Yusof R, Bahaman N, Mostafa SA, Foozy CFM. Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. IEEE Access. 2021;9:22351–70. doi:10.1109/ACCESS.2021.3056614.
37. Elhanashi A, Gasmi K, Begni A, Dini P, Zheng Q, Saponara S. Machine learning techniques for anomaly-based detection system on CSE-CIC-IDS2018 dataset. In: Proceedings of the International Conference on Applications in Electronics Pervading Industry, Environment and Society; 2022 Sep; Cham, Switzerland. p. 131–40. doi:10.1007/978-3-031-30333-3_17.
38. Singh R, Srivastav G. Novel framework for anomaly detection using machine learning technique on CIC-IDS2017 dataset. In: Proceedings of the 2021 International Conference on Technological Advancements and Innovations (ICTAI); 2021 Nov; Hyderabad, India. p. 632–6. doi:10.1109/ICTAI53825.2021.9673238.
39. Chen Y, Liu X, Rao M, Qin Y, Wang Z, Ji Y. Explicit speed-integrated LSTM network for non-stationary gearbox vibration representation and fault detection under varying speed conditions. Reliab Eng Syst Saf. 2025;254:110596. doi:10.1016/j.ress.2024.110596.
40. Ahmed S, Nielsen IE, Tripathi A, Siddiqui S, Ramachandran RP, Rasool G. Transformers in time-series analysis: a tutorial. Circuits Syst Signal Process. 2023;42(12):7433–66. doi:10.1007/s00034-023-02454-8.
41. Nasayreh A, Alawad NA, Jaradat A. Enhanced chimp optimization algorithm using crossover and mutation techniques with machine learning for IoT intrusion detection system. Clust Comput. 2025;28(7):455. doi:10.1007/s10586-025-05119-0.