# A Two-Stage Hybrid Intrusion Detection Framework Based on Hierarchical Attack Mapping and Pruned CNN-GRU Models

**Aseel M. Mohammed**
Ministry of Higher Education and Scientific Research, Scientific Research Commission, Iraq
aseel-phdcs@mohesr.edu.iq (corresponding author)

**Haider K. Hoomod**
Computer Department, College of Education, Al-Mustansiriyah University, Ministry of Higher Education and Scientific Research, Baghdad, Iraq
drhjnew@gmail.com

*Received: 11 November 2025 | Revised: 25 December 2025 and 27 December 2025 | Accepted: 29 December 2025*

 *DOI: https://doi.org/10.48084/etasr.16210*

**ABSTRACT**

**This study presents a realistic two-step approach to network intrusion detection that combines two established security paradigms, namely, anomaly-based and signature-based detection. The proposed hybrid architecture achieves high detection accuracy and maintains system customizability and scalability for real-world applications, even for resource-limited edge devices. The proposed method first compares all network traffic packets with a large resource of known attack signatures (data-driven signature file), which is generated from actual data from network attacks, helping in faster detection of known threats. Packets that do not match this signature verification are then processed to a more advanced analytical step, where a specialized CNN-GRU hybrid model takes over. This model was optimally pruned, significantly reducing computational costs and inference delays but still allowing it to identify attacks without adversely affecting system throughput. To ensure strict evaluation, six high-profile benchmark datasets, namely NSL-KDD, UNSW-NB15, BCCC, CIC-UNSW-NB15, NF-ToN-IoT-v3, and CICIOT2023, were aligned under a single feature schema. In addition, a hierarchical attack taxonomy was designed, where on the simplest level a binary classification (Normal or Attack) is performed, followed by the classification of general attack types and, lastly, the fine-grained classification of particular attack forms. Each dataset was used to train a dedicated and pruned CNN-GRU model. For inference, an advanced voting system is used to combine the predictions of all constituent models, producing a much-trusted determination of network activity. Across both binary and multi-class evaluations, the system achieves up to 99.99% accuracy, with high F1-scores between 94% and 100%. This high accuracy did not come at the cost of speed, as the pruning process notably reduced computational overhead and sped up analysis. Its modular architecture allows the system to be easily adapted with new datasets or even directly analyze live network traffic, making it a robust and scalable solution for modern cybersecurity challenges. Unlike existing intrusion detection approaches that suffer from critical limitations, such as dependence on a single-stage anomaly detection, training on limited data, and relying on complex designs that hinder their scalability and increase computational cost, the proposed two-stage pruned CNN-GRU architecture with hierarchical attack mapping is capable of overcoming these limitations and maintaining high detection accuracy while reducing computational overhead.**

***Keywords-two-stage hybrid NIDS; pruned deep learning; CNN-GRU architecture; network cybersecurity***

## I. INTRODUCTION

Network Intrusion Detection Systems (NIDS) are critical for the protection of the ever-growing interrelated ecosystem of modern cyber infrastructure [1]. In the last ten years, systems have gradually adopted deep learning methods to detect a range of adversarial activities. However, this trend faces three major challenges. First, models trained on a single benchmark dataset, such as NSL-KDD or UNSW-NB15, tend not to identify intrusions based on slightly different environments and, therefore, tend to inhibit their ability to generalize in the heterogeneous environment of operational networks. Second, the heterogeneity of feature representations from different research datasets prevents the development of a single, powerful defensive model. Third, complex deep learning models that deliver higher detection rates demand prohibitive computing resources. Although previous studies have tried to alleviate these shortcomings, some gaps remain substantial. To

better understand these challenges, the following section reviews relevant studies.

### *A. Literature Review*

In [2], the validation of a high-performance GPU-assisted CNN-GRU model was limited to a project-specific combination of Mirai and Bashlite botnet traffic based on the N-BaloT dataset, achieving 0.8946 accuracy and 0.8807 F1-score. This architecture follows a single-stage anomaly-based paradigm and does not address the inherent computational complexity and generalization induced by narrow testing horizons. In [3], a single-stage CNN-GRU model achieved high accuracy (99.52%) and F1-score (99.49%), which makes it theoretically adequate to use in the context of IoT applications. However, its applicability is confined because it relies on the data specific to the IoT healthcare domain without generalization to a wider range of network ecosystems. In [4], CNN and GRU were combined with the Convolutional Block Attention Model (CBAM) across standard data sets (NSL-KDD, UNSW-NB15). The best configuration, the parallel CNN-CBAM-GRU, achieved very high accuracy, reaching 99.56% on NSL-KDD. Despite this high performance, the model complexity from the added CBAM component and a dual-path fusion strategy makes it computationally demanding.

This trend of complex architectures is also evident in other studies. In [5], a CNN-LSTM-GRU framework was presented for IoT and IIoT security, where the CNN output is fed into parallel LSTM and GRU branches, and their temporal features are then merged for final classification. This framework is a single-stage anomaly-based detector, but its intricate design leads to high computational cost, limiting its practical scalability. In addition, its validation was restricted to only the ToN-IoT and CICIDS2017 datasets, narrowing the scope of its proven robustness. In [6], a single-stage NIDS integrated Bidirectional Short-Term Memory Modules (BiLSTM) and GRUs. Although this approach yielded a high accuracy of 99.86% on the ToN-IoT dataset, it resulted in a large-scale model, which potentially hinders its deployment in scenarios that require real-time analysis or limited computational resources. The CNN-GRU-FF model [7] is a single-stage NIDS that employs double-layer feature fusion and a modified focal loss function, achieving a detection rate of 99.68% on the NSL-KDD dataset. However, being evaluated within a limited data scope, this model exemplifies the generalization problem that plagues the field. The limited scope was also observed in [8], using only two datasets, UNSW-NB15 and BoT-IoT, to evaluate a CNN-GRU detection model that achieved 93% accuracy. However, the key novelty was the application of the Self-Upgraded Cat and Mouse Optimization (SUCMO) algorithm to fine-tune the model's hyperparameters and enhance classification accuracy, while the core architecture is simple. It should be noted that SUCMO makes the training phase complex and potentially challenging to reproduce. In [9], high accuracies (99.95% and 99.76%) were achieved on the KDDCup99 and NSL-KDD datasets, respectively. The proposed model was implemented on a GPU-enabled platform using a parallel CNN-GRU. The features extracted by these two parallel streams were merged through horizontal concatenation to capture both the spatial and sequential attributes of the traffic. However, the validation was limited to classic and specific datasets.

Data preprocessing techniques can sometimes artificially inflate model performance, compromising the validity of results for real-world deployment. The model in [10] achieved high accuracies of 99.83% and 99.01% on the IoTID20 and BoT-IoT datasets, respectively, but employed the SMOTE balancing technique before splitting the data into training and test sets. This fundamental methodological approach means the model was tested on synthetic data that it was already indirectly exposed to during training. Consequently, the reported high accuracy (99.83% and 99.01%) does not reflect true performance on genuine, unseen attacks, making it unreliable for real-world deployment. In addition, the use of methods like SMOTE is also likely to increase the latency of training and increase computational cost, thus compromising the need for lightweight deployable solutions. In [11], a CNN-GRU-based Multi-Dimensional Latent (MDL) architecture was specifically designed for IoT intrusion detection. This one-stage anomaly-based intrusion detection system achieved a high accuracy of 99.60%. However, this study used the FWSMOTE augmentation method, which, although it allegedly addresses the issue of class imbalance, introduces more complications.

In [12], an attention-based hybrid CNN-GRU architecture was proposed to identify intrusions, with a detailed preprocessing pipeline. This pipeline combines the ADRDB algorithm, which is a synthesis of ADASYN, RENN, and DBSCAN, to equalize the proportion of classes, and the RFP algorithm, which uses Random Forests with Pearson correlation to isolate salient features. However, this study admitted that the model had only a slight improvement in the detection of minority attack classes. This tradeoff highlights one of the underlying limitations, as preprocessing complexity does not necessarily eliminate the inherent problem of detecting rare attacks but instead multiplies computational complexity, therefore undermining the viability of the system.

### *B. Problem Definition*

Based on the review of the state-of-the-art literature, the deployment of high-performance NIDS is hindered by the following challenges:

- Current NIDS models often use single-stage anomaly detection based on complex designs such as attention mechanisms or multi-branch structures. This leads to high computational costs, making them unsuitable for use in resource-limited IoT applications. This necessitates a new NIDS that not only maintains a high level of detection but, at the same time, boasts of simplicity and, above all, exhibits a high level of generalizability to diverse network settings. To address these challenges, this study suggests a two-phase hybrid IDS model, which is pragmatic.
- A major hurdle for most NIDS research is poor generalization. Models are often trained and tested on just one or two benchmark datasets, failing to account for the significant differences in how data is represented across various network environments. Furthermore, the use of incorrect data handling practices, such as pre-split oversampling, does not reflect true performance on unseen

attacks, making these practices unreliable for real-world deployment.

### C. Contributions

The principal contributions of this work are as follows.

1. A hybrid two-stage architecture combines both fast signature-based filtering and a deep learning-based anomaly detector. The two-stage design allows for the early filtering of known malicious traffic, reducing the number of samples processed by the second-stage CNN-GRU model. As a result, the detection will cover both known and new attacks, reduce resource consumption, offering faster response times compared to single-stage approaches.
2. Uses a systematic harmonization of features and a hierarchical attack mapping to focus on generalizability with six different sets of benchmark data. This improves the robustness and performance of the model, using data from different sources.
3. A lean CNN-GRU network is strategically pruned through an optimized design to realize efficiency. Such optimization results in a substantial decrease in the complexity of computation and inference time without loss of high detection accuracy at the level needed to deploy in the real world.
4. A decision mechanism is added through a voting-based ensemble of several models. This ensemble approach ensures that there is coherence in the performance of various network environments.
5. The effectiveness and practical utility of the tool are supported by comprehensive empirical validation in the form of both unified and heterogeneous datasets with a detection rate of up to 99.99-1 and a high F1-score.

This work, filling the gap between accuracy and operational efficiency, provides a solid and scalable framework that can be deployed to implement modern NIDS.

## II. PROPOSED METHOD AND SYSTEM DESIGN

The proposed framework integrates three key strategies: harmonizing data from diverse sources, employing a two-stage detection process, and utilizing a streamlined, pruned neural network.

### A. Foundation: Multi-Dataset Harmonization

To ensure that the pruned CNN-GRU model is robust and generalized to an extensive range of network settings, the initial move was to determine a unified data structure. Six datasets, namely NSLKDD, UNSWNB15, CICNSUNB15, BCCC packet CloudDDoS2024, CICIoT2023, and NFToNIoTv3, were structurally unified by a strict standardization protocol. Preprocessing involved careful data cleaning, where duplicate entries and infinite values were removed, and missing numerical values were imputed through feature-specific median replacement. This was then followed by feature-space standardization to align ubiquitous network features, including IP addresses, port numbers, and flow duration, with standard nomenclature. As an illustration, lexical column names such as src ip and Source IP Address were brought into the same identifier, as was the case with the use of numbers (e.g., 6 transformed into a descriptive term such as TCP). This step was taken to ensure that each dataset had a feature vector whose ordering was the same. The hierarchical classification system was implemented to reduce discrepancies in labeling attacks, using a three-layer framework:

- Level 1 (Binary): The simplest classification, distinguishing between Normal or Attack.
- Level 2 (Multi-class): Grouping specific attacks into general categories such as "DoS/DDoS" or "Reconnaissance" to enhance model generalization by minimizing rare classes.
- Level 3 (Original): Retaining the original, highly specific attack name (e.g., SQL Injection) for detailed analysis and insight.

After preprocessing and dataset harmonization, a reduced subset of the six datasets was used. Finally, a global LabelEncoder was applied across all datasets to ensure consistent numerical encoding for categorical features. As a result, six harmonized datasets were produced, which share identical names for common features, appearing in the same order in the datasets, with common data types and attack taxonomy.

The preprocessing of the aggregated datasets was performed before model training. In order to reduce the dominance of features with large numerical ranges, the Min-Max scaling limited all values to the unit interval [0, 1]. The data were then divided into 70% for training, 15% for validation, and 15% for testing.

### B. Pruned CNN-GRU Model Preparation and Training

A hybrid CNN-GRU architecture, which combines the spatial discriminative ability of CNNs and the temporal modelling effectiveness of GRUs, is the heart of the anomaly detection engine. This bilateral-nature structure maintains the fine-grained feature extraction and can track dynamic patterns across the sequence, therefore providing a powerful detection ability. The model was implemented in PyTorch. It started with a 1-D convolutional layer with 64 filters of size 3 to detect the local spatial patterns and a max-pooling operation (kernel size 2, stride 2). The combined outputs were subsequently input into a GRU containing 32 hidden units to acquire time-based dependencies. The feature maps were again refined with a second convolutional step (32 filters), which had an associated max-pooling and then flattening. This was then followed by two fully-connected layers (64 and 32 units) with a dropout layer (rate 0.2) in front of them to reduce overfitting.

One of the salient innovations is the use of global unstructured pruning on all the trainable weights of the CNN, GRU, and dense layers. The pruning process was applied after the full training phase and before the inference stage. The preliminary experiments included testing different pruning ratios (50%, 60%, 70% and 80%). The 70% ratio was selected

because it achieved the best balance between model reduction and minimal accuracy loss. This process uses an L1-norm criterion to find and eliminate 70% of the least important weight parameters, which significantly decreases the size and complexity of the model and speeds up the inference process. This resulted in the final architecture being highly appropriate to implementation in resource-constrained settings. Algorithm 1 outlines the pruning process.

```
Algorithm 1: Global Unstructured Pruning
Input: Model M, Pruning Amount α (e.g., 0.7)
Output: Pruned Model M'
Identify all prune-eligible parameters P in M
(Weights/Biases of Conv, GRU, Linear layers).
Collect all parameter values into a single
global vector Wglobal.
Determine the magnitude threshold T
corresponding to the α-th percentile of the L1
norms of Wglobal.
Apply a mask Mmask where Mmask = 0 if |w|<T and
Mmask=1 otherwise
Apply the mask Mmask globally to all parameters
P.
Return M'.
```

For training and optimization, the output layer utilizes the Softmax activation function with cross-entropy loss for multi-class classification (Level 2), while the sigmoid activation function with binary cross-entropy loss is employed for Level 1 classification. All model training utilizes the Adam optimizer with a default learning rate of 0.01 over 15 epochs and a batch size of 64. Table I presents the layer-by-layer configuration of the pruned hybrid CNN-GRU model.

TABLE I. PRUNED HYBRID CNN-GRU ARCHITECTURE

| # | Layer type | Input shape | Output shape | Pruning status |
|---|---|---|---|---|
| 0 | Conv1D | (N,1, L) | (N,64, L) | Pruned (70%) |
| 1 | ReLU | (N,64, L) | (N,64, L) | N/A |
| 2 | MaxPool1D | (N,64, L) | (N,64, L`) | N/A |
| 3 | Reshape (Permute) | (N,64, L`) | (N, L`,64) | N/A |
| 4 | GRU Layer | (N, L`,64) | (N, L`,32) | Pruned (70%) |
| 5 | Reshape (Permute) | (N, L`,32) | (N, 32, L`) | N/A |
| 6 | Conv1D | (N, 32, L`) | (N, 32, L`) | Pruned (70%) |
| 7 | ReLU | (N, 32, L`) | (N, 32, L`) | N/A |
| 8 | MaxPool1D | (N, 32, L`) | (N, 32, L″) | N/A |
| 9 | Flatten | (N, 32, L″) | (N, M) | N/A |
| 10 | Dense1 | (N, M) | (N, 64) | Pruned (70%) |
| 11 | ReLU + Dropout(0.2) | (N, 64) | (N, 64) | N/A |
| 12 | Dense2 | (N, 64) | (N, 32) | Pruned (70%) |
| 13 | ReLU + Dropout(0.2) | (N, 32) | (N, 32) | N/A |
| 14 | Output Layer | (N, 32) | (N,1) or (N, C) | Pruned (70%) |
| 15 | Sigmoid (Binary) / Softmax (Multi-class) | (N,1) or (N, C) | (N,1) or (N, C) | N/A |

## C. Two-Stage Hybrid NIDS Architecture

Figure 1 shows the sequential process of the two-stage hybrid architecture for NIDS. It combines a fast signature-based filter with a sophisticated deep learning analyzer to identify both known and novel threats and ensure comprehensive security coverage.

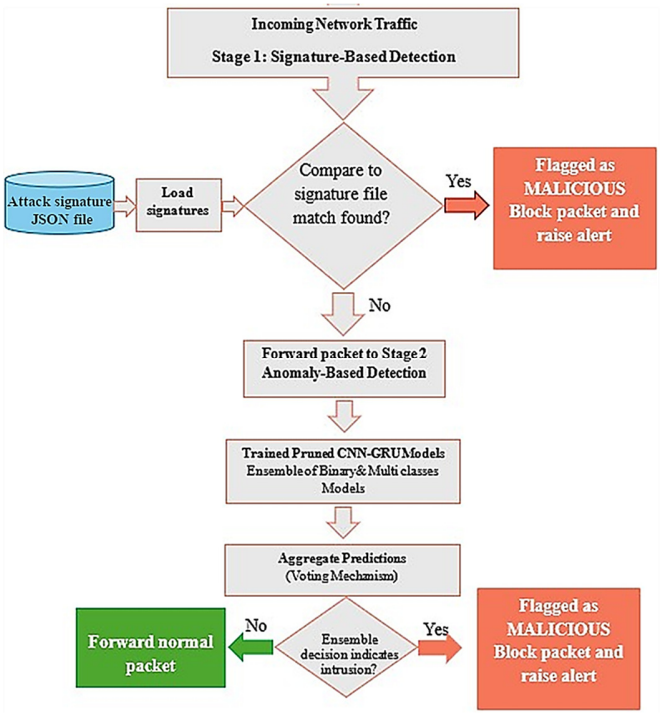


Fig. 1. Execution flow of the two-stage hybrid NIDS.

### 1) Stage 1: Signature-Based Fast Filtering

This stage is a high-speed security checkpoint. This work generated a custom signature file extracted from actual data from network attacks in the CICIOT2023 dataset. This dataset is distributed across several files; this study used four files (Merged45-Merger48) and enforced column unification across them, because of inconsistent feature names, to build the signature file. This process ensures that the signatures used accurately represent attack behavior. When network traffic comes in, important attributes, such as protocol, port numbers, and TCP/UDP flags, are compared to the signature database. This ensures faster detection and reduces the load on the anomaly detection model because only unknown or suspicious packets will be forwarded to the second stage.

### 2) Stage 2: Pruned CNN-GRU Ensemble for Anomaly Detection

Traffic filtered through the initial filter is sent to the core detection engine, which is a collection of pruned CNN-GRU models. All samples pass through the same preprocessing pipeline as that used during training to offer uniformity and strength to the ultimate detection phases. After the processed sample is evaluated by the multiple specialized CNN-GRU models, individual decisions are combined through majority voting, where the most frequently predicted label becomes the final classification. The system outputs both a simple normal/attack verdict and the specific category identification (such as DoS, Scanning, or Exploits), providing security teams with both immediate alerts and detailed context.

## III. EXPERIMENTAL DATASETS

The lack of sufficiently diverse and challenging test data is one of the main problems on the evaluation of any NIDS. The proposed hybrid model was evaluated using six separate datasets, specifically selected to cover a wide range of network settings, traffic patterns, and modern cyber-attacks. The NSL-KDD, a historical benchmark, provides a common baseline on which comparative analysis can be performed [13]. The UNSW-NB15 [14] and CIC-UNSW-NB15 [15] datasets were included to measure the performance of the proposed framework in the complexity of contemporary networks. The suite also involves the use of specialized corpora, i.e., BCCC-cPacket-Cloud-DDoS-2024, a large-scale Cloud-DDoS construct [16], CIC-IoT-2023 [17], and NF-ToN-IoT-v3 [18], which involve IoT-specific intrusions, to determine the model's ability to process voluminous NetFlow traffic. Table II tabulates salient differences across these datasets, including scale, granularity, and diversity of attacks.

TABLE II. PROPERTIES AND MAJOR ATTACK CLASSES OF THE EXPERIMENTAL DATASETS

| Dataset | Records | Features | Attack categories |
|---|---|---|---|
| NSL-KDD | 148517 | 42 | DoS, Probe (Scanning), R2L (Remote-to-Local), U2R (User-to-Root), Normal |
| UNSW-NB15 | 2059413 | 49 | DoS, Exploits, Fuzzers, Reconnaissance, Generic, Analysis, Backdoor, Shellcode, Worms, Normal |
| CIC-UNSW-NB15 | 2350508 | 55 | DoS, Exploits, Fuzzers, Reconnaissance, Generic, Analysis, Backdoor, Shellcode, Worms, Benign |
| BCCC-cPacket-Cloud-DDoS-2024 | 700776 | 324 | Cloud-based DDoS attacks (various TCP flood types), Suspicious traffic, Benign |
| CIC-IoT-2023 | 877337 | 47 | IoT-based attacks (DoS, DDoS, Mirai variants, Reconnaissance, Vulnerability Scan, Web attacks), Benign |
| NF-ToN-IoT-v3 | 6426030 | 55 | DoS/DDoS, Injection, Scanning, Password/Brute-force, XSS, Backdoor, MITM, Ransomware, Benign |

## IV. EXPERIMENTAL RESULTS AND DISCUSSION

The performance of the proposed pruned hybrid CNN-GRU model was evaluated across six different benchmark datasets in performing two key security tasks: distinguishing attacks from normal traffic (binary classification) and identifying specific types of attacks (multi-class classification). Tables III and IV summarize the standard performance metrics for binary and multi-class classification across the datasets.

Tables V and VI provide a direct comparison of key metrics (Accuracy and F1-score) from the studies discussed in the literature review subsection and the methodological advantages of the proposed work, focusing on its broader generalization capability and more efficient computational design compared to previously published studies.

TABLE III. BINARY CLASSIFICATION RESULTS

| Dataset | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| NSL-KDD | 99% | 99% | 99% | 99% |
| BCCC-cPacket-Cloud-DDoS-2024 | 94% | 95% | 94% | 94% |
| CIC-UNSW-NB15 | 100% | 100% | 100% | 100% |
| CIC-IoT-2023 | 99% | 99% | 99% | 99% |
| UNSW-NB15 | 99% | 99% | 99% | 99% |
| NF-ToN-IoT-v3 | 94% | 94% | 94% | 94% |

TABLE IV. MULTI-CLASS CLASSIFICATION RESULTS

| Dataset | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|
| NSL-KDD | 98% | 98% | 98% | 98% |
| BCCC-cPacket-Cloud-DDoS-2024 | 91% | 91% | 91% | 90% |
| CIC-UNSW-NB15 | 98% | 98% | 98% | 98% |
| CIC-IoT-2023 | 98% | 98% | 98% | 97% |
| UNSW-NB15 | 98% | 98% | 98% | 98% |
| NF-ToN-IoT-v3 | 93% | 93% | 93% | 93% |

TABLE V. MODEL PERFORMANCE COMPARISON

| Study | Binary | | Multi-class | |
|---|---|---|---|---|
| | Accuracy | F1-score | Accuracy | F1-score |
| Proposed | 94-100% | 94-100% | 91-98% | 90-98% |
| [2] | N/A | N/A | 89.46% | 88.07% |
| [3] | 99.52% | 99.49% | N/A | N/A |
| [4] | 99.27-99.57% | N/A | 96.3-99.56% | N/A |
| [5] | 99.50-100% | 99.49-100% | 97% | 99.94% |
| [6] | 98.69-99.86% | 99.01- 99.92% | N/A | N/A |
| [7] | N/A | N/A | 99.54-99.86% | 98.28-98.68% |
| [8] | 92-93% | N/A | N/A | N/A |
| [9] | 99.78% | 99.7% | 99.63% | **99.6%** |
| [10] | N/A | N/A | 99.98% | 99.98% |
| [11] | N/A | N/A | 99.60% | 99.61% |
| [12] | 99.96% | 99.96% | 99.73% | 99.72% |

TABLE VI. METHODOLOGY AND EFFICIENCY COMPARISON

| Study | Datasets | Model Arch. |
|---|---|---|
| Proposed | NSL-KDD, UNSW-NB15, CIC-UNSW, CICIOT2023, BCCC-2024, NF-ToN-IoT | Pruned CNN-GRU with global pruning (70%) for inference time reduction + Maximized generalization (via 6 harmonized datasets |
| [2] | Customized IIoT datasets from N-BaloT | CNN-GRU+GPU-Assisted Acceleration |
| [3] | IoT Healthcare Security dataset | Inherently lightweight architecture of CNN-GRU (5 trainable layers) for IoT resource constraints. |
| [4] | UNSW-NB15, NSL-KDD | CNN-CBAM Attention Mechanism -GRU (Parallel Fusion) |
| [5] | TON_IoT CICIDS2017 | CNN-LSTM-GRU (Multi- branch parallel - sequential fusion arch.) + SMOTE |
| [6] | CSE-CICIDS2018, ToN_IoT, | GRU and BiLSTM |
| [7] | UNSW-NB15, NSL-KDD | CNN-GRU-FF (Double-layer Feature Fusion) + Modified Focal Loss Function |
| [8] | UNSW-NB15, BoT-IoT | GRU-CNN + (SUCMO) algorithm |
| [9] | NSL-KDD | Parallel CNN-GRU+ GPU-accelerated |
| [10] | IoTID20, BoT-IoT | CNN-GRU+ SMOTE |
| [11] | IoTID20, UNSW-NB15 | CNN-GRU + FW-SMOTE (Feature-weighted synthetic minority oversampling technique) |
| [12] | UNSW_NB15, NSL-KDD, CIC-IDS2017 | Sequential CNN–GRU |

## V. CONCLUSION

This study presented a two-stage hybrid intrusion detection framework that addresses three critical issues in modern NIDS, namely, suboptimal generalization, lack of standardization, and high-computational costs. By performing systematic data harmonization along with a categorical attack taxonomy, the proposed framework demonstrates a high level of performance under heterogeneous network conditions.

Central to the proposed framework is a simplified CNN-GRU design, which achieves state-of-the-art detection performance with state-of-the-art accuracy values as high as 99.99% and F1-scores between 94 and 100%, with significant improvements in computational efficiency. Using a global pruning algorithm to cut 70% of the model parameters, a significant computational reduction was achieved with no detectable loss in the fidelity of the detection task. The two-level architecture, which combines rapid signature filtering with a simplified deep-learning ensemble, is an essential and realistic framework that can be deployed in real-time in resource-limited environments. Future work involves an automated labeling system that will make the framework more flexible to new, emergent, and unseen attack patterns, following the constant development of cyber threats.

## REFERENCES


[1] X. Li, Z. Zheng, M. Zhao, Y. Zhao, L. Shi, and B. Wang, "RLFE-IDS: A framework of Intrusion Detection System based on Retrieval Augmented Generation and Large Language Model," *Computer Networks*, vol. 268, Aug. 2025, Art. no. 111341, https://doi.org/10.1016/j.comnet.2025.111341.

[2] C. P. R. Rani and K. Baalaji, "A graphics processing unit assisted CNN-GRU framework for the intrusion detection mechanism in the industrial internet of things," *Engineering Research Express*, vol. 7, no. 2, Feb. 2025, Art. no. 025240, https://doi.org/10.1088/2631-8695/adc971.

[3] A. Usman, "Enhancing Cybersecurity in IoT Healthcare Systems: A CNN-GRU Hybrid Approach for Intrusion Detection," M.S. Thesis, Dublin, National College of Ireland, 2025.

[4] S. M. Hosseini, A. Ebrahimi, M. R. Mosavi, and H. Sh. Shahhoseini, "A novel hybrid CNN-CBAM-GRU method for intrusion detection in modern networks," *Results in Engineering*, vol. 28, Dec. 2025, Art. no. 107103, https://doi.org/10.1016/j.rineng.2025.107103.

[5] D. M. A. A. Afraji, J. Lloret, L. Peñalver, D. M. A. A. Afraji, J. Lloret, and L. Peñalver, "An Integrated Hybrid Deep Learning Framework for Intrusion Detection in IoT and IIoT Networks Using CNN-LSTM-GRU Architecture," *Computation*, vol. 13, no. 9, Sept. 2025, https://doi.org/10.3390/computation13090222.

[6] A. A. Ghani and S. A. Alasadi, "A Deep Learning Algorithm to Cybersecurity: Enhancing Intrusion Detection with a Hybrid GRU and BiLSTM Model," *Engineering, Technology & Applied Science Research*, vol. 15, no. 3, pp. 23605–23612, June 2025, https://doi.org/10.48084/etasr.10666.

[7] Y. Imrana, Y. Xiang, L. Ali, A. Noor, K. Sarpong, and M. A. Abdullah, "CNN-GRU-FF: a double-layer feature fusion-based network intrusion detection system using convolutional neural network and gated recurrent units," *Complex & Intelligent Systems*, vol. 10, no. 3, pp. 3353–3370, June 2024, https://doi.org/10.1007/s40747-023-01313-y.

[8] A. Sagu, N. S. Gill, P. Gulia, N. Alduaiji, P. K. Shukla, and M. A. Shah, "Advances to IoT security using a GRU-CNN deep learning model trained on SUCMO algorithm," *Scientific Reports*, vol. 15, no. 1, May 2025, Art. no. 16485, https://doi.org/10.1038/s41598-025-99574-9.

[9] W. Chen, "Intelligent Network Intrusion Detection for Advanced Measurement System Based on CNN-GRU Modeling," *International Journal of Network Security*, vol. 27, no. 1, Jan. 2025, https://doi.org/10.6633/IJNS.202501_27(1).16.

[10] K. O. Adefemi, M. B. Mutanga, O. A. Alimi, K. O. Adefemi, M. B. Mutanga, and O. A. Alimi, "A Hybrid CNN–GRU Deep Learning Model for IoT Network Intrusion Detection," *Journal of Sensor and Actuator Networks*, vol. 14, no. 5, Sept. 2025, https://doi.org/10.3390/jsan14050096.

[11] A. Qaddos, M. U. Yaseen, A. S. Al-Shamayleh, M. Imran, A. Akhunzada, and S. Z. Alharthi, "A novel intrusion detection framework for optimizing IoT security," *Scientific Reports*, vol. 14, no. 1, Sept. 2024, Art. no. 21789, https://doi.org/10.1038/s41598-024-72049-z.

[12] B. Cao *et al.*, "Network Intrusion Detection Model Based on CNN and GRU," *Applied Sciences*, vol. 12, no. 9, Apr. 2022, https://doi.org/10.3390/app12094184.

[13] L. Dhanabal and S. P. Shantharajah, "A study on NSL-KDD dataset for intrusion detection system based on classification algorithms," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 4, no. 6, pp. 446–452, 2015.

[14] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *2015 Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia, Nov. 2015, pp. 1–6, https://doi.org/10.1109/MilCIS.2015.7348942.

[15] "UNSW-NB15 Augmented Dataset." Canadian Institute for Cybersecurity, [Online]. Available: https://www.unb.ca/cic/datasets/cic-unsw-nb15.html.

[16] M. Shafi *et al.*, "Toward Generating a New Cloud-Based Distributed Denial of Service (DDoS) Dataset and Cloud Intrusion Traffic Characterization," *Information*, vol. 15, no. 4, Mar. 2024, https://doi.org/10.3390/info15040195.

[17] E. C. P. Neto *et al.*, "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors*, vol. 23, no. 13, June 2023, https://doi.org/10.3390/s23135941.

[18] "NF-ToN-IoT." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/dhoogla/nftoniot.