JCTA Journal of Computing Theories and Applications ISSN:3024-9104

Research Article

An Attention-Enhanced CNN–RBF Framework for Network Intrusion Detection in Imbalanced Traffic

Fabrice Kabura [1,*] and Thierry Nsabimana [2]

[1] Data Science Program, African Institute for Mathematical Sciences (AIMS-Senegal), Mbour-Thiès 23000, Sénégal; e-mail : kabura.fabrice@aims-senegal.org
[2] Institute of Applied Statistics (ISTA), University of Burundi, Bujumbura, Burundi; e-mail : thierry.nsabimana1984@gmail.com
* Corresponding Author : Fabrice Kabura

Abstract: The increasing complexity and scale of modern network traffic driven by IoT and cloud-based infrastructures have made accurate intrusion detection a critical challenge. Conventional network intrusion detection systems (NIDS) and many deep learning–based approaches struggle to reliably detect minority and stealthy attacks due to severe class imbalance and limited discrimination of subtle traffic patterns. To address these limitations, this study proposes a hybrid CNN–RBF–Attention framework for network intrusion detection. The proposed model integrates three complementary components: (i) a convolutional neural network for hierarchical feature extraction from network flow data, (ii) a radial basis function (RBF) network for localized nonlinear classification using prototype-based decision regions, and (iii) an attention mechanism that adaptively weights RBF activations to emphasize discriminative traffic patterns. SMOTE is applied exclusively to the training data to mitigate class imbalance. The framework is evaluated on the widely used CICIDS2017 and CICIDS2018 benchmark datasets in both binary and multiclass settings, using recall, precision, F1-score, confusion matrices, and ROC analysis. Experimental results demonstrate that the proposed hybrid model consistently outperforms standalone CNN and RBF baselines, particularly in terms of recall and F1-score. On the CICIDS2018 dataset, the model achieves 99.81% accuracy and 99.81% F1-score in binary classification, and 99.54% accuracy and 99.54% F1-score in multiclass classification. On CICIDS2017, it achieves 98.12% accuracy and 98.12% F1-score in binary classification, and 98.92% accuracy and 98.92% F1-score in multiclass classification. Confusion matrix and ROC analyses further show strong class separability and reliable performance in low–false-positive-rate regions, which is critical for real-world IDS deployment. These results confirm that combining deep hierarchical feature learning, localized prototype-based classification, and attention-guided refinement yields a robust, operationally reliable intrusion detection framework for highly imbalanced network environments.

Keywords: Anomaly Detection; Attention Mechanism; CNN–RBF Model; Cybersecurity Analytics; Hybrid Deep Learning; Internet of Things Security; Network Intrusion Detection; Network Traffic Analysis.

Received: December, 29th 2025
Revised: January, 28th 2026
Accepted: January, 29th 2026
Published: January, 31st 2026

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) licenses (https://creativecommons.org/licenses/by/4.0/)

1. Introduction

The rapid expansion of interconnected infrastructure driven by technologies such as the Internet of Things (IoT), 5G, fog computing, utility computing, grid computing, edge computing, and cloud services has dramatically increased both the complexity and volume of network traffic[1]. This surge has enabled cyber-attacks to evolve into more advanced, covert, and persistent threats, leaving modern networks highly susceptible to incidents such as ransomware, zero-day exploits, distributed denial-of-service (DDoS) attacks, and man-in-the-middle attacks [2]–[4].

Network Intrusion Detection Systems (NIDS) serve as a critical defense layer, continuously monitoring network traffic to identify and block unauthorized access and malicious activity. Conventional approaches, which depend on signature-based matching, can

effectively detect known threats but struggle against new or highly sophisticated attacks. They also tend to produce excessive false positives and lack the precision and speed needed to handle today's dense, high-velocity traffic streams [5], [6].

To address these shortcomings, the cybersecurity research community has shifted toward intelligent NIDS that incorporate Machine Learning (ML) and Deep Learning (DL) methods. Techniques such as K-Means, Logistic Regression, Decision Trees, Support Vector Machines (SVM), Recurrent Neural Networks (RNN), and Convolutional Neural Networks (CNN) excel at processing massive datasets, recognizing attack patterns, and uncovering subtle anomalies that evade signature rules [7]–[11]. Incorporating ML and DL has markedly boosted detection accuracy, precision, and the ability to spot previously unknown threats. Nevertheless, to our knowledge, no study has investigated a hybrid framework that fuses CNNs with an Attention-Enhanced Radial Basis Function (ARBF) network for cybersecurity intrusion detection.

Motivated by this gap, this work proposes a hybrid CNN–RBF–Attention framework for network intrusion detection. The proposed model combines:

- Deep Convolutional Neural Networks (DCNNs) for hierarchical feature extraction, enabling rich representations of complex traffic patterns;
- Radial Basis Function (RBF) networks for prototype-based nonlinear classification, allowing localized and fine-grained separation between benign and malicious flows; and
- An attention mechanism applied to RBF activations, which adaptively emphasizes the most relevant prototypes, improving discrimination in imbalanced and ambiguous attack scenarios.

The main objective of this research is to design and evaluate a hybrid security framework that enhances intrusion detection performance in complex network environments. Specifically, the proposed approach aims to improve detection accuracy and recall across diverse attack types, reduce false-positive rates through prototype-based decision modeling, and enhance generalization to previously unseen threats while maintaining a modular, scalable architecture. Section 2 reviews related work and identifies key research gaps in existing intrusion detection approaches. Section 3 presents the proposed CNN–RBF–Attention framework, including the model architecture, training strategy, and preprocessing pipeline. Section 4 reports the experimental setup, evaluation protocol, and results on the CICIDS2017 and CICIDS2018 datasets, followed by comparative analysis and discussion. Finally, Section 5 concludes the paper by summarizing the main findings, discussing limitations, and outlining directions for future research.

2. Theoretical Background and Related Works

2.1. CNN-based Feature Extraction for NIDS

Convolutional Neural Networks (CNNs) are widely used in NIDS because they can automatically learn hierarchical feature representations from high-dimensional traffic data. In the proposed framework, a deep CNN is employed as an automated feature extractor to transform raw or selected network traffic features into compact and discriminative representations that subsequent classifiers can effectively process [12]–[14]. Figure 1 illustrates the CNN architecture used for feature extraction. The network consists of an input layer with feature dimensions matching the selected features, followed by successive convolutional and pooling layers that capture local temporal patterns and higher-level abstractions from network flow sequences. Fully connected layers with dropout are included to improve generalization and reduce overfitting.

The CNN is built with TensorFlow/Keras, using convolutional, pooling, and dense layers. The architecture includes an input layer, multiple Conv1D layers (with 32–128 filters, kernel size 3, and ReLU activation), MaxPooling1D layers for downsampling, and dense layers with dropout regularization. An output softmax layer is used during pretraining but is not employed as the final decision component in the hybrid model. Instead, the CNN primarily functions as a hierarchical feature extractor. Feature vectors are extracted from the penultimate dense layer, resulting in embeddings of dimension $d = 128$, which are then passed to the RBF classifier for final intrusion detection.

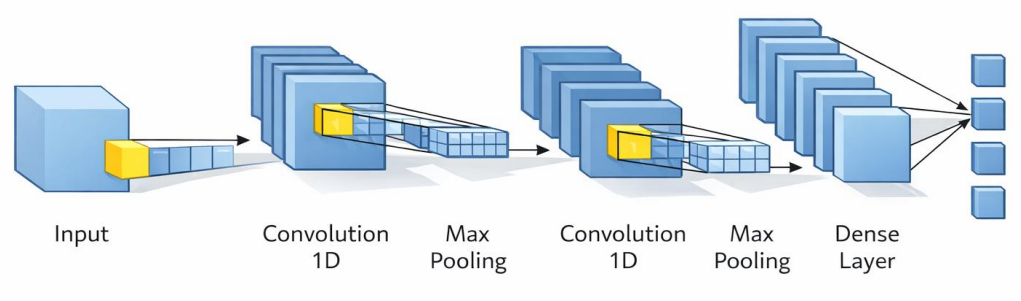

Figure 1. Conceptual illustration of a CNN-based feature extraction process (adapted for illustration purposes).

To extract hierarchical representations from network traffic sequences, the CNN applies a series of convolution and pooling operations that progressively transform the input into higher-level feature maps. For each convolutional layer l, the feature map produced by filter f at position i is computed as:

$$Z^{[l]}[f,i] = max(\cdot) \tag{1}$$

where a kernel size of $K = 3$ and a stride $S = 1$, are used, with zero-padding applied to preserve the spatial dimensions of the input. The output length after convolution is given by:

$$H_{out} = \left[\frac{H + 2P - K}{S}\right] + 1 \tag{2}$$

where H denotes the input length, and P is the padding size. To reduce dimensionality while retaining the most informative patterns, max pooling is applied to the resulting feature maps:

$$P[i] = \max_{0 \le m < K_p} Z^{[l]}\left[i + S_p + m\right] \tag{3}$$

with a pooling window size $K_p = 2$ and stride $S_p = 2$. After successive convolution and pooling stages, the output of the final dense layer (before the softmax activation) is flattened into a feature vector $f \in R^d$ which serves as the input to the subsequent RBF classifier.

Through this sequence of operations, the CNN automatically learns local temporal patterns and increasingly abstract feature interactions from network traffic data, producing compact and discriminative representations that significantly enhance intrusion detection performance when combined with the RBF-based classification module.

2.2. RBF-based Classification for Intrusion Detection

Radial Basis Function (RBF) networks are widely used for nonlinear classification due to their ability to model localized decision boundaries in high-dimensional feature spaces. In the proposed framework, the RBF network serves as the final classification component, operating on the high-level feature representations extracted by the CNN. By mapping these features into an RBF space, the classifier enables precise discrimination between normal and malicious traffic patterns [15]–[17]. Figure 2 illustrates the RBF architecture employed in this study. The network consists of an input layer, a hidden layer of Gaussian radial basis neurons, and a linear output layer. Each hidden neuron measures the similarity between the input feature vector and a learned prototype (center), allowing the model to capture localized variations in network traffic behavior that are often difficult to separate using global decision boundaries.

Formally, for each RBF unit $k = 1, \dots, K$, the activation of the hidden layer is computed as:

$$\phi_k(x) = exp\left(-\frac{\|x - \mu_k\|^2}{2\sigma^2}\right) \tag{4}$$

where $\mu_k \in \mathbb{R}^d$, denotes the center obtained via K-Means clustering on the CNN feature space, $\sigma > 0$ is the spread parameter, and $x \in \mathbb{R}^d$ is the input feature vector. The output logits for the j-th class are then computed as:

$$Z_j = \sum_{k=1}^{N} W_{kj}\phi_k(x) + b_j, j = 1,2,\dots,C \tag{5}$$

where $W_{kj} \in \mathbb{R}$ and b_j are the learned weights and bias parameters of the output layer. These logits are transformed into normalized class probabilities using the softmax function:

$$\hat{y}_j = \frac{e^{z_j}}{\sum_{c=1}^{C} e^{z_c}}, j = 1,2,\dots,C \tag{6}$$

and the final predicted class is obtained as:

$$\hat{y}_{final} = \arg\max_i \hat{y}_j \tag{7}$$

By combining K-Means–initialized centers with a trainable linear output layer, the RBF classifier efficiently captures localized nonlinear decision boundaries in the CNN-transformed feature space. This design enhances interpretability, accelerates convergence, and improves classification performance, particularly for overlapping or minority attack classes. The integration of the RBF classifier with hierarchical features extracted from CNNs enables effective intrusion detection by combining deep representation learning with prototype-based nonlinear classification.

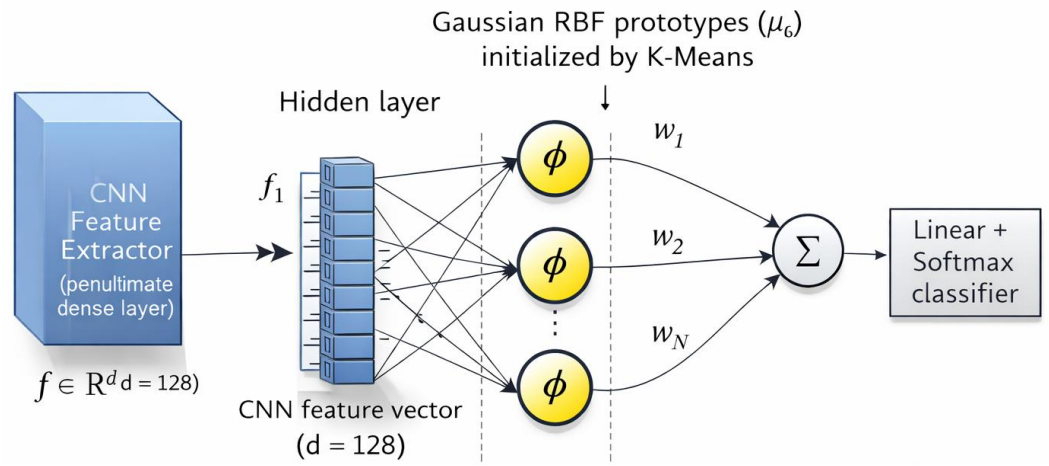

Figure 2. Conceptual illustration of an RBF-based classifier applied to CNN-extracted feature embeddings.

2.3. Related Works and Research Gap

In recent years, numerous studies have explored hybrid models for network intrusion detection, evaluated on benchmark datasets such as CICIDS2017, CICIDS2018, UNSW-NB15, and KDD Cup 99. These works include ensemble-based machine learning and deep learning approaches designed to improve robustness under multiclass imbalanced intrusion detection scenarios [18].

For instance, Qazi et al. [19] proposed a convolutional recurrent neural network (CNN–RNN) hybrid model for intrusion detection, leveraging CNN for local feature extraction and RNN for sequential analysis. Evaluated on CICIDS2018, the model achieved an average accuracy of 98% in detecting malicious attacks. While effective for handling temporal dependencies in network traffic, the model lacks an attention mechanism, limiting its ability to focus on rare or zero-day attacks, and exhibits limited cross-dataset generalization due to dataset-specific tuning.

Umair et al. [7] introduced a hybrid multilayer deep learning model combining a multilayer CNN for feature extraction and selection with a softmax classifier, supplemented by a deep neural network for classification. Tested on the NSL-KDD and KDDCUP'99 datasets,

this approach achieved 99% accuracy. Although it excels at high-level feature abstraction from large datasets, it struggles with nonlinear class separation for closely related attack types and lacks adaptability to emerging threats, as it relies heavily on predefined statistical patterns rather than dynamic feature weighting.

Zhao and Zhao [20] developed a hybrid RBF–SVM model using RBF networks for feature extraction and Support Vector Machines (SVM) for intrusion detection. On the KDD99 dataset, the model achieved 97% accuracy and over 99% precision. The RBF component provides strong nonlinear mapping capabilities; however, the model suffers from high computational demands when applied to large-scale datasets and demonstrates limited effectiveness in handling highly imbalanced classes or zero-day attacks without additional mechanisms for feature prioritization.

More recently, Sajid et al. [21] presented a hybrid ML–DL approach integrating XGBoost and CNN for feature extraction, combined with LSTM for classification. Evaluated on CICIDS2017, UNSW-NB15, NSL-KDD, and WSN-DS datasets, the model demonstrated high detection rates with low false acceptance rates. Despite its robustness across multiple datasets, this ensemble-based approach introduces high computational complexity due to the integration of boosting and recurrent layers, along with limited interpretability, a common limitation in large-scale bagging-based IDS frameworks [22], which can hinder real-time deployment and analysis of rare attack patterns.

Overall, these studies demonstrate that hybrid intrusion detection models can achieve strong performance on specific benchmark datasets. However, several limitations remain consistently observed across prior work:

- Limited generalization across datasets or real-world traffic conditions due to dataset-specific optimization strategies.
- Insufficient nonlinear separation of closely related or rare attack classes, particularly in models lacking advanced kernel functions or adaptive mechanisms.
- Difficulty adapting to zero-day or emerging attacks, as many approaches lack mechanisms for focusing on subtle anomalies.
- High computational complexity and limited interpretability, especially in ensemble-heavy or recurrent hybrid architectures.

Despite the strong performance of recent CNN and deep learning–based intrusion detection systems, several challenges remain insufficiently addressed. Pure CNN or DL architectures typically rely on global feature representations and softmax-based decision boundaries, which are sensitive to class imbalance and often fail to capture subtle local variations associated with rare or emerging attack patterns. In highly imbalanced IDS datasets such as CICIDS2017 and CICIDS2018, these models frequently achieve high overall accuracy while exhibiting poor recall for minority attack classes [23]. Similarly, RBF-based models and prototype-driven classifiers offer strong local decision-making capabilities and interpretable nonlinear mappings [24], as demonstrated by recent hybrid RBF-based intrusion detection frameworks [20]. However, when used in isolation, these models depend heavily on the quality of handcrafted or pre-extracted features and lack hierarchical representation learning.

Recent studies have shown that attention mechanisms can significantly enhance intrusion detection performance, particularly in highly imbalanced scenarios where minority and hard-to-detect attack classes are easily overlooked [18], [25]. However, most existing hybrid approaches still rely on combinations of deep models and classical classifiers without integrating attention-guided prototype weighting. As a result, these models rarely unify deep hierarchical feature extraction, prototype-based local classification, and adaptive attention mechanisms within a single framework, limiting their ability to focus on rare or subtle attack patterns.

3. Proposed Method and Experimental Setup

3.1. Overview of the Proposed Hybrid CNN–RBF–Attention Framework

To overcome the limitations of existing intrusion detection approaches in handling highly imbalanced traffic, overlapping attack patterns, and rare or emerging threats, we propose a hybrid CNN–RBF–Attention model that integrates deep feature extraction, nonlinear prototype-based classification, and adaptive attention within a unified pipeline. The overall

workflow of the proposed method is illustrated in Figure 3, which presents the main processing stages from data acquisition and preprocessing to final intrusion prediction.

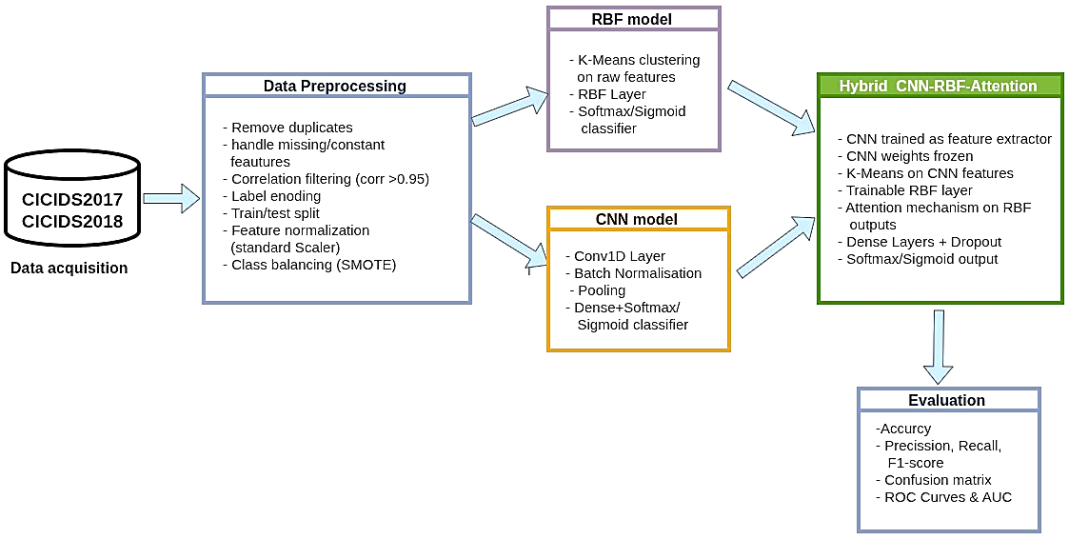

Figure 3. Intrusion detection model workflow for a hybrid model, add an attention mechanism

As shown in Figure 3, the pipeline consists of three core components. First, a CNN-based feature extraction module is used to learn hierarchical representations from network traffic data. By employing a one-dimensional convolutional architecture optimized for sequential flow features, the CNN captures local temporal patterns and higher-order feature interactions, producing compact embeddings that serve as inputs to subsequent classification stages.

Second, the extracted CNN feature embeddings are passed to an RBF layer, which performs nonlinear classification in a prototype-based feature space. The RBF layer maps deep feature representations into an RBF space using Gaussian prototypes initialized via K-Means clustering, enabling localized decision boundaries that are particularly effective for separating closely related or minority attack classes.

Third, an attention mechanism is applied to the RBF activations to weight the most informative prototypes for each input sample dynamically. This attention-guided weighting allows the model to focus on discriminative basis functions while suppressing less relevant responses, thereby enhancing sensitivity to subtle and low-frequency attack patterns.

This modular yet unified design combines the representation-learning strengths of deep neural networks with the interpretability and local discrimination capabilities of prototype-based classifiers, while avoiding the computational complexity of ensemble-heavy or recurrent architectures. Detailed descriptions of the dataset characteristics and preprocessing steps are provided in Section 3.2, and the architectural configuration and training procedure of the hybrid model are presented in Section 3.3.

3.2. Datasets and Preprocessing

The proposed hybrid CNN–RBF–Attention model is evaluated using two widely adopted benchmark datasets for network intrusion detection: CICIDS2017 and CICIDS2018. These datasets differ in scale, traffic composition, and attack diversity, providing a comprehensive basis for assessing model robustness and generalization. The CICIDS2018 dataset contains 6,659,532 network flow records described by 80 traffic features, including flow-based statistics, packet counts, and protocol-specific attributes. It includes a wide range of attack types such as DDoS, brute-force, port scanning, botnet activity, and benign traffic. The CICIDS2017 dataset consists of 2,830,744 flow records with a similar feature representation, covering attack categories such as DDoS, infiltration, brute-force, and web-based attacks, and capturing diverse temporal and behavioral patterns of network activities [23].

Table 1 reports the original class distributions of both datasets before any preprocessing or resampling. As shown, both datasets exhibit severe class imbalance, with benign traffic accounting for more than 70% of samples in CICIDS2017 and approximately 80% in

CICIDS2018. Several attack categories account for less than 0.1% of total traffic, reflecting the highly skewed nature of real-world intrusion detection data and motivating the use of imbalance-aware training strategies.

Table 1. Original class distribution of CICIDS2017 and CICIDS2018 datasets.

CICIDS2017 Class	Samples	%	CICIDS2018 Class	Samples	%
BENIGN	2,273,097	72.3	Benign	5,329,008	80.0
DoS Hulk	231,073	7.3	DDoS LOIC-HTTP	575,364	8.6
PortScan	158,930	5.1	DDoS HOIC	198,861	3.0
DDoS	128,027	4.1	DoS Hulk	145,199	2.2
DoS GoldenEye	10,293	0.33	Bot	144,535	2.2
FTP-Patator	7,938	0.25	Infiltration	118,483	1.8
SSH-Patator	5,897	0.19	SSH-Bruteforce	94,048	1.4
DoS Slowloris	5,796	0.18	DoS GoldenEye	41,406	0.6
DoS Slow-httptest	5,499	0.17	DoS Slowloris	9,908	0.15
Bot	1,966	0.06	DDoS LOIC-UDP	1,730	0.03
Web Attack BF	1,507	0.05	Brute Force-Web	568	<0.01
Web Attack XSS	652	0.02	Brute Force-XSS	229	<0.01
Infiltration	36	<0.01	SQL Injection	85	<0.01
Web Attack SQLi	21	<0.01	DoS Slow-HTTPTest	55	<0.01
Heartbleed	11	<0.01	FTP-BruteForce	53	<0.01

To ensure data quality, consistency, and computational efficiency, comprehensive preprocessing was applied. Duplicate records were removed, missing or inconsistent values were handled, and highly correlated features (correlation coefficient > 0.95) were filtered out to reduce redundancy. Numerical features were normalized using StandardScaler, while categorical attributes were encoded using OneHotEncoding to ensure compatibility with both CNN and RBF components.

To prevent model performance from being dominated by extremely rare classes, six minority attack categories with very limited samples (Brute Force-XSS, SQL Injection, Infiltration, Brute Force-Web, DoS SlowHTTPTest, and FTP-BruteForce) were excluded from selected multiclass experiments. For completeness and fair comparison with previous studies, binary classification experiments (benign vs. attack) were also conducted on the full datasets without any class exclusion. Finally, to avoid data leakage, SMOTE was applied exclusively to the training set after the train–test split. The test set was kept fully untouched and retained the original class distribution, ensuring unbiased and reliable performance evaluation.

3.3. Hybrid CNN–RBF–Attention Model: Architecture and Training

This section presents the architecture and training procedure of the proposed hybrid CNN–RBF–Attention model. The framework integrates a one-dimensional CNN for hierarchical feature extraction, a prototype-based RBF layer for nonlinear classification, and an attention mechanism for adaptive prototype weighting. The complete architecture is illustrated in Figure 4, and the detailed hyperparameter configuration is summarized in Table 2.

3.3.1. CNN-Based Feature Extraction

The CNN component is designed as a one-dimensional convolutional network tailored to sequential network flow features. It consists of stacked Conv1D layers, followed by batch normalization, max pooling, and dropout to ensure stable training and effective regularization. The convolutional layers use 32, 64, and 128 filters with kernel size 3 and ReLU

activation, enabling the extraction of local temporal patterns and higher-order feature interactions from network traffic data.

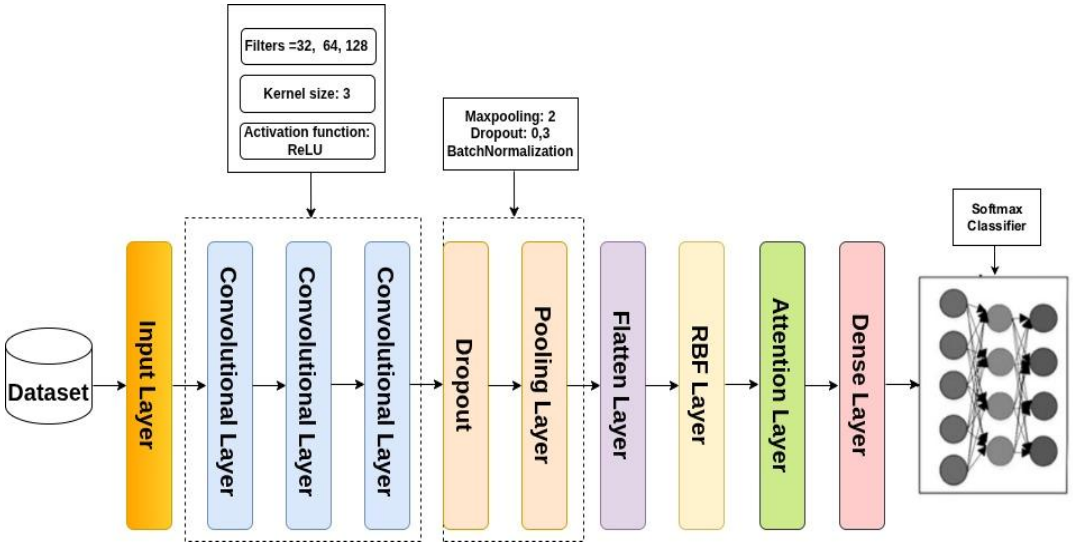

Figure 4. CNN-RBF-Attention learning framework

Table 2. Architectural hyperparameters of the proposed hybrid CNN-RBF-attention model

Component	Parameter / Configuration	Value / Description
CNN Feature Extractor	Architecture	Custom CNN (Conv1D → MaxPooling → Conv1D → MaxPooling → Flatten)
	Input shape	(216000, 9, 1)
	Number of filters	32 → 64 → 128
	Kernel sizes	3, 3
	Pooling	MaxPooling1D (pool size = 2)
RBF Layer	Number of centers (K)	128
	Center initialization	K-Means clustering on 10,000 training CNN features
	Trainable centers	No (fixed after initialization)
	Gamma initialization	1 / (2 × median nearest-neighbor distance²)
	Gamma training	Trainable (softplus-parameterized)
Attention Mech-anism	Type	Lightweight residual gating attention (multiplicative)
	Attention heads	Single-head
	Weight matrix dimension	128 × 128
	Learnable	Yes (end-to-end)
Classifier Head	Optimizer	Adam (learning rate = 1e−4)
	Loss function	Categorical / Binary Cross-Entropy
	Regularization	Early stopping (patience = 3, monitor = val_loss)

Given an input sequence $X \in \mathbb{R}^{L\times 1}$, the output of the convolutional layer at position i is computed as:

$$Z_i^{[l]} = \sigma\left(\sum_{k=1}^{K} w_k \cdot X_{i+k-1} + b\right) \tag{8}$$

where w_k denotes convolutional kernel weights, b is the bias term, K is the kernel size, and $\sigma(\cdot)$ is the ReLU activation function. Max pooling is applied to reduce dimensionality while retaining dominant features.

After training, the final softmax classifier is discarded, and feature embeddings are extracted from the penultimate dense layer, yielding a compact representation $f \in \mathbb{R}^d$ (with $d = 128$). These embeddings serve as inputs to the RBF layer.

3.3.2. RBF Layer with Prototype Initialization

The RBF layer performs nonlinear classification on CNN-extracted feature embeddings using Gaussian basis functions. A total of $K = 128$ RBF units are employed, each representing a prototype in the embedding space. The centers $\mu_k \in \mathbb{R}^d$ are initialized using K-Means clustering on a random subset of 10,000 normalized training embeddings, ensuring representative coverage of the feature distribution. The activation of the k-th RBF unit is defined as:

$$\phi_k(f) = exp(-\gamma \|f - \mu_k\|^2) \tag{9}$$

where γ is the spread parameter controlling the width of the Gaussian basis function. The initial value of γ is set using the median nearest-neighbor distance between cluster centers:

$$\gamma_{\text{init}} = \frac{1}{2\sigma^2}, \qquad \sigma = \text{median}\left(\|\mu_i - \mu_j\|_2\right) \tag{10}$$

To ensure numerical stability, γ is optimized during training using a softplus-parameterized scalar.

3.3.3. Attention Mechanism and Classification Head

To enhance sensitivity to informative and minority attack patterns, a lightweight multiplicative attention mechanism is applied to the RBF activations. Attention weights are computed as:

$$\alpha = \text{softmax}(W_a \phi + b_a) \tag{10}$$

where $W_a \in \mathbb{R}^{d \times d}$ and b_a are learnable parameters. The attended RBF representation is obtained by element-wise multiplication:

$$\tilde{\phi} = \phi \odot \alpha \tag{11}$$

This mechanism emphasizes discriminative prototypes while suppressing less informative ones. The attended representation is then passed through two fully connected layers with ReLU activation, batch normalization, and dropout (rates 0.2 and 0.3). The final output layer applies softmax (for multiclass tasks) or sigmoid (for binary tasks) to produce class probabilities.

3.3.4. Training Strategy and Optimization

Training is performed in two stages to ensure stable convergence and effective representation learning. In the first stage, the CNN feature extractor is trained independently using cross-entropy loss to learn robust hierarchical representations. After convergence, CNN weights are frozen, and feature embeddings are extracted from the penultimate layer.

In the second stage, the RBF layer, attention mechanism, and dense classification head are trained using the extracted embeddings. The model is optimized using the Adam optimizer with a learning rate of 10^{-4} . Early stopping is applied based on validation loss, with a patience of 3 epochs, to prevent overfitting. Categorical cross-entropy is used for multiclass classification, while binary cross-entropy is applied for binary tasks.

3.3.5. Inference and Confidence-Based Fusion

During inference, the hybrid model produces class probabilities from the attention-enhanced RBF classifier. To further reduce false positives, the output of the hybrid classifier can be combined with the standalone CNN softmax output using a weighted fusion scheme:

$$\hat{y}_{\text{final}} = \arg\max_i \left(\lambda \hat{y}_i^{\text{CNN}} + (1 - \lambda)\hat{y}_i^{\text{hybrid}}\right) \tag{12}$$

where $\lambda \in [0,1]$ is tuned on a validation set to balance detection sensitivity and false positive rate.

For clarity and reproducibility, the overall training and inference pipeline of the proposed model is summarized in Algorithm 1.

Algorithm 1. Training and Inference Procedure of the Proposed CNN–RBF–Attention Model

INPUT: Network flow features X, labels y
OUTPUT: Predicted class $\hat{y}$
1: Train CNN using cross-entropy loss to learn hierarchical feature representations.
2: Extract feature embeddings $f \in \mathbb{R}^d$ from the penultimate dense layer.
3: Normalize embeddings using training-set statistics.
4: Initialize RBF centers μ_k via K-Means clustering.
5: Compute RBF activations $\phi_k(f)$ using Gaussian basis functions.
6: Apply attention to RBF activations to obtain weighted responses.
7: Train dense classification head using cross-entropy loss.
8: During inference, optionally fuse CNN and hybrid outputs to reduce false positives.
9: Output final prediction $\hat{y}$.

3.4. Experimental Setup

This section describes the experimental design used to evaluate the proposed hybrid CNN–RBF–Attention model. It covers the evaluation protocol, baseline models, ablation settings, and the computational environment used to ensure reproducibility and fair comparison. All experiments are conducted on the CICIDS2017 and CICIDS2018 benchmark datasets. For stability in multiclass evaluation, several attack categories with extremely small sample sizes are excluded, as listed in Section 3.2. These exclusions are applied consistently across all models, ensuring that comparisons are conducted under identical class-selection conditions.

The performance of the proposed approach is evaluated on both binary classification (benign vs. attack) and multiclass classification tasks, reflecting realistic intrusion detection scenarios. Three main model families are considered: a standalone CNN model, a standalone RBF model, and the proposed hybrid CNN–RBF–Attention model. All models are trained and evaluated using the same preprocessing pipeline, including normalization, encoding, and class balancing, to ensure fairness. To assess the contribution of individual components, an ablation study is performed.

The CNN-only model uses the same convolutional backbone followed by a softmax classifier, serving as a deep learning baseline. The RBF-only model operates on handcrafted and preprocessed features without CNN-based representation learning. Additional hybrid variants include CNN–RBF without attention and CNN–RBF with attention. This design enables a clear assessment of the impact of CNN feature extraction, RBF-based nonlinear classification, and the attention mechanism. The results show that the attention-enhanced hybrid model significantly improves recall for minority attack classes while maintaining low false positive rates.

3.4.1. Environment Setup

All experiments are conducted in a cloud-based computing environment using Google Colab, which provides virtual machines with dynamic resource allocation. Each experimental session typically uses a standard runtime configuration consisting of 2 CPU cores (Intel® Core™ i7-8650U @ 1.90 GHz × 8), up to 16 GB of RAM, and a Tesla T4 GPU for accelerated training when available. Approximately 256 GB of disk space is allocated for dataset storage and model checkpoints.

The implementation is based on Python (version 3.10+) and relies on a comprehensive set of open-source libraries. TensorFlow (v2.15+) and Keras are used to build and train deep learning models. Pandas and NumPy support data manipulation and preprocessing. Scikit-learn is employed for feature scaling (e.g., StandardScaler, LabelEncoder), evaluation metrics (e.g., classification report, confusion matrix, ROC–AUC), and class balancing using SMOTE from the imbalanced-learn library. Visualization of experimental results is performed using Matplotlib and Seaborn. To ensure reproducibility, all datasets are downloaded programmatically via the Kaggle API, and a fixed random seed (SEED = 42) is used for all stochastic operations, including data splitting, sampling, and model initialization.

3.5. Evaluation Protocol and Performance Metrics

The performance of the proposed models is evaluated using a controlled, consistent protocol designed for intrusion detection tasks with highly imbalanced class distributions. All experiments follow the training and inference procedure described in Algorithm 1, and identical preprocessing, data splits, and class selection criteria are applied to all compared models to ensure fairness. The datasets are divided into 80% training and 20% testing sets using stratified sampling to preserve the original class distribution. Model selection and early stopping are performed using a validation subset drawn solely from the training data. Performance is reported on the held-out test set, which remains fully untouched throughout training.

3.5.1. Confusion Matrix and Class-Wise Evaluation

To analyze classification behavior in detail, a confusion matrix summarizes the relationship between predicted and true labels, including true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). This analysis is particularly important for intrusion detection, where false negatives result in undetected attacks and pose a critical security risk.

3.5.2. Primary Evaluation Metrics

Several complementary metrics are used to provide a comprehensive evaluation of intrusion detection performance. Given the highly imbalanced nature of network traffic data, no single metric is sufficient; different measures are interpreted with varying priorities. Accuracy is reported to provide a global view of overall classification correctness. However, in highly imbalanced intrusion detection datasets, high accuracy can be achieved by favoring the dominant benign class [9], [26]. Therefore, accuracy is not used as the primary performance indicator and is interpreted with caution.

$$\text{accuracy} = \frac{TP + TN}{TP + FN + TN + FP} \tag{13}$$

Recall measures the proportion of attacks correctly detected and is treated as the primary evaluation metric, particularly for minority and rare attack classes, where false negatives are most critical.

$$\text{recall} = \frac{TP}{TP + FN} \tag{13}$$

Precision quantifies the reliability of detected attacks and complements recall by controlling false alarms.

$$\text{precision} = \frac{TP}{TP + FP} \tag{13}$$

The F1-score provides a balanced summary of detection performance by combining precision and recall.

$$\text{F1} = \frac{2 \times \text{recall} \times \text{precision}}{\text{precision} + \text{recall}} \tag{13}$$

For multiclass experiments, macro-averaged metrics are reported to ensure that all attack classes contribute equally to the evaluation, preventing dominant classes from masking poor detection on minority categories.

3.5.3. ROC Analysis and False Positive Behavior

Receiver Operating Characteristic (ROC) curves are used to analyze the trade-off between true positive rate (TPR) and false positive rate (FPR) across different decision thresholds. The Area Under the ROC Curve (AUC) provides a threshold-independent measure of class separability [27], [28]. In security-sensitive scenarios, particular attention is given to low-FPR operating regions, where intrusion detection systems are typically deployed to avoid overwhelming analysts with false alerts. Models that maintain high recall while operating at low false positive rates are considered more suitable for real-world deployment.

3.5.4. Reporting Strategy

Results are reported for both binary (benign vs. attack) and multiclass classification tasks. For multiclass evaluation, per-class recall is used to highlight performance on minority attack

categories, while macro-averaged metrics summarize overall detection effectiveness. This evaluation protocol ensures that reported performance reflects not only overall accuracy but also the model's ability to detect rare and high-risk intrusions under realistic, imbalanced conditions.

4. Results and Discussion

This section presents the experimental results of the proposed hybrid CNN–RBF–Attention model on the CICIDS2018 and CICIDS2017 datasets under both binary and multiclass classification settings. Performance is compared against two baselines: a standalone CNN and a standalone RBF network. All models are evaluated using the same preprocessing pipeline, data splits, and evaluation protocol described in Section 3.

Given the highly imbalanced nature of intrusion detection datasets, recall and macro-averaged recall are emphasized as the primary evaluation metrics, while accuracy is reported for completeness. Special attention is paid to performance in low false-positive-rate (FPR) regions, which are critical for real-world IDS deployment.

4.1. Results on CICIDS2018 Dataset

The CICIDS2018 dataset contains more than six million network flows described by 78 initial features and exhibits severe class imbalance, posing a significant challenge for reliable intrusion detection. To mitigate this issue, SMOTE oversampling and label encoding were applied to the training data for multiclass classification. In addition, feature selection was performed to retain only the most discriminative variables, reducing noise and computational overhead and improving model generalization.

To ensure stable learning and avoid bias introduced by extremely rare classes, several attack categories with very limited samples were excluded, namely Brute Force-XSS, SQL Injection, Brute Force-Web, DoS SlowHTTPTest, and FTP-Brute Force. The Infiltration class was also omitted due to its well-documented ambiguity and high confusion with benign traffic reported in prior studies. This preprocessing strategy maintains realistic learning conditions while balancing class representativeness and robustness.

Table 3 summarizes the binary and multiclass classification results obtained on the CICIDS2018 dataset. The standalone RBF model provides a baseline but struggles with overlapping attack distributions and minority classes. The CNN model substantially improves performance by learning hierarchical representations, achieving higher recall across most attack categories. However, residual misclassifications remain for low-frequency attacks, indicating the limitation of global feature representations.

Table 3. Classification results on CICIDS2018

Class	Model	Acc. (%)	Prec. (%)	Rec. (%)	F1 (%)
Binary classification	RBF	98.92	98.96	98.92	98.63
	CNN	99.63	99.64	99.63	99.63
	CNN-RBF	99.74	99.74	99.74	99.74
	CNN-RBF-Attention	99.81	99.81	99.81	99.81
Multiclass classification	RBF	96.59	96.63	96.59	96.56
	CNN	99.21	99.21	99.21	99.21
	CNN–RBF	99.49	99.50	99.49	99.49
	CNN–RBF–Attention	99.54	99.55	99.54	99.54

The proposed CNN–RBF–Attention model achieves the strongest performance across all metrics, with consistent improvements in recall and F1-score while maintaining a low false positive rate. The improvement is particularly evident for minority and hard-to-detect attack classes, confirming that integrating localized RBF decision regions with attention-based weighting enhances sensitivity to subtle and rare attack patterns.

To analyze misclassification behavior in detail, only the confusion matrices of the proposed model are presented. Figure 6 shows the multiclass confusion matrix for CICIDS2018, which exhibits strong diagonal dominance across all attack categories. Misclassifications between closely related DoS variants and between benign and low-rate attacks are substantially

reduced, indicating that the attention mechanism effectively prioritizes discriminative RBF prototypes derived from CNN features. The ROC curves of the proposed model are shown in Figure 7. Although near-perfect AUC values are observed across most classes due to the dataset's overall separability, the proposed model shows a steep increase in the true positive rate at low false–positive rates. This operating characteristic is critical for real-world IDS deployment, where even small increases in false alarms can lead to significant operational overhead.

For binary classification, the confusion matrix in Figure 8 shows that the proposed model achieves near-perfect separation between benign and attack traffic, with only a minimal number of misclassifications. The corresponding ROC curve in Figure 9 further confirms superior sensitivity in low-FPR regions, reinforcing the suitability of the proposed model for deployment in practical intrusion detection systems where recall and false alarm control are more critical than accuracy alone.

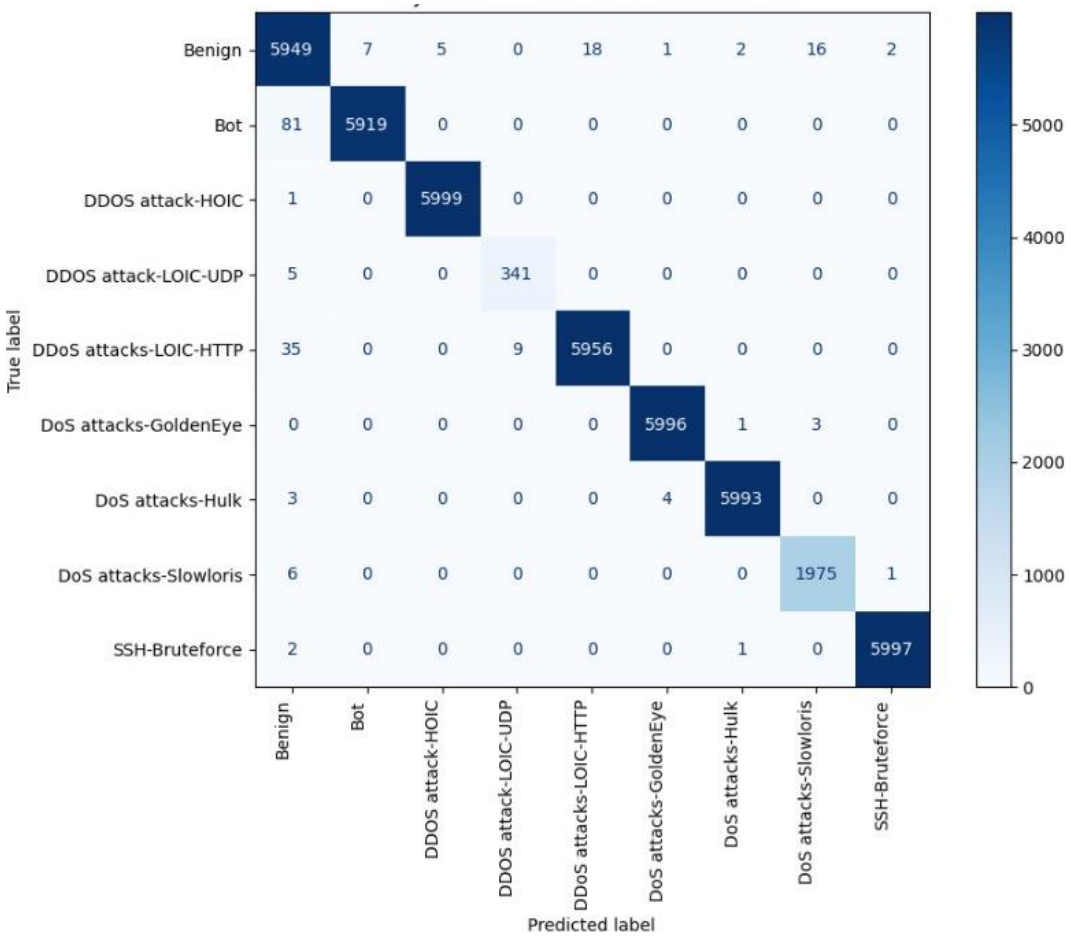

Figure 6. Multiclass confusion matrix for CICIDS2018 results

The ROC curves (Figures 8 and 9) show that although all models achieve near-perfect AUC values due to the dataset's overall separability, the proposed model shows a steeper rise in the true positive rate at low false–positive rates. This operating regime is particularly critical for real-world IDS deployment, where even small increases in false alarms can overwhelm security operations. The results on CICIDS2018 indicate that the proposed hybrid architecture not only improves overall classification performance but also enhances robustness under extreme class imbalance, making it more suitable for practical intrusion detection scenarios where minority attack detection and controlled false alarms are primary requirements.

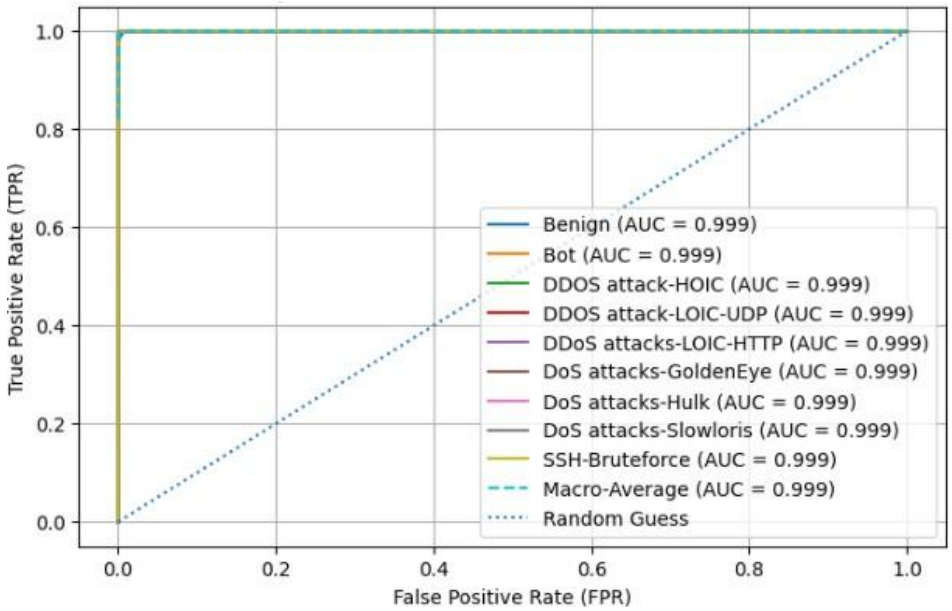

Figure 7. Multiclass ROC-AUC for CICIDS2018 results

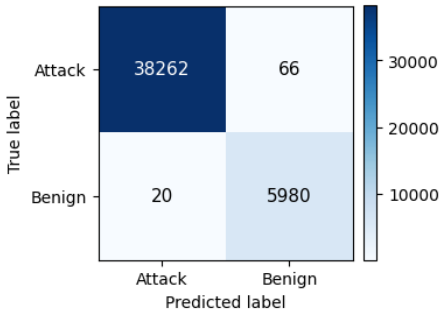

Figure 8. Binary class confusion matrix for CICIDS2018 results

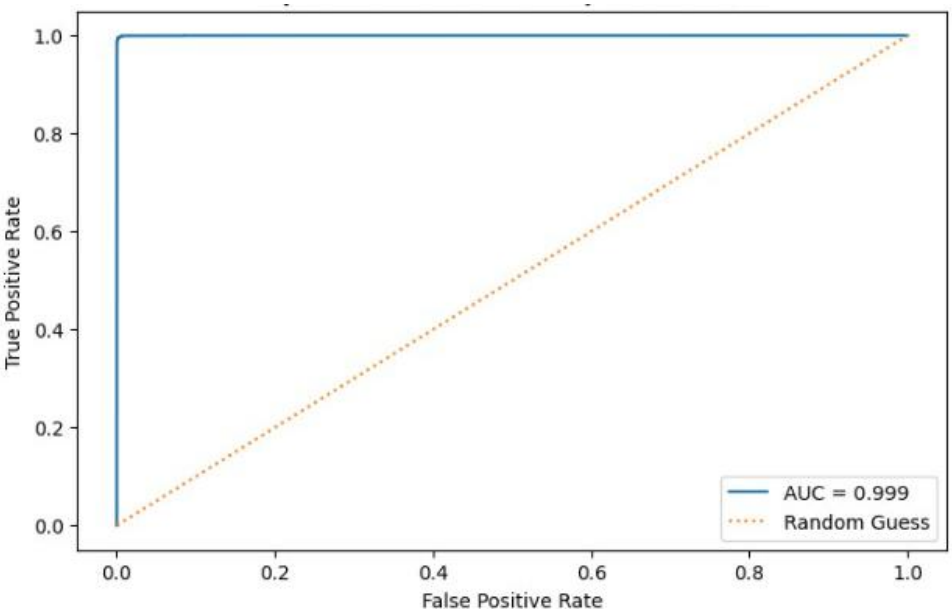

Figure 9. Binary class ROC-AUC for CICIDS2018 results

4.2 Results on CICIDS2017 Dataset

The CICIDS2017 dataset represents a more challenging intrusion detection scenario than CICIDS2018 due to its higher diversity of attack types, stronger temporal variations, and

more severe class imbalance across multiple low-frequency categories. The same preprocessing pipeline and evaluation protocol described in Section 3 were applied to ensure fair and consistent comparison across all models. Binary and multiclass classification results are reported in Table 4.

Table 4. Classification results on CICIDS2017

Class	Model	Acc. (%)	Prec. (%)	Rec. (%)	F1 (%)
Binary classification	RBF	92.57	92.73	92.57	92.56
	CNN	97.86	97.87	97.86	97.86
	CNN-RBF	97.97	97.98	97.97	97.97
	CNN-RBF-Attention	98.12	98.12	98.12	98.12
Multiclass classification	RBF	91.66	91.81	91.66	91.41
	CNN	97.89	97.94	97.89	97.68
	CNN–RBF	98.18	98.20	98.18	98.18
	CNN–RBF–Attention	98.92	98.95	98.92	98.92

As shown in Table 4, the standalone RBF classifier provides a baseline level of performance but suffers from elevated false negatives when distinguishing overlapping and low-rate attack classes. This limitation is particularly evident in multiclass settings, where prototype-based classification without hierarchical feature abstraction struggles to separate subtle attack behaviors from benign traffic. The CNN model significantly improves overall performance by learning hierarchical and correlation-aware feature representations, achieving strong recall for high-volume attacks. However, its performance remains sensitive to severe class imbalance, as reflected by reduced recall for minority attack classes. This behavior indicates that global deep representations alone are insufficient to capture localized variations in rare intrusion patterns fully.

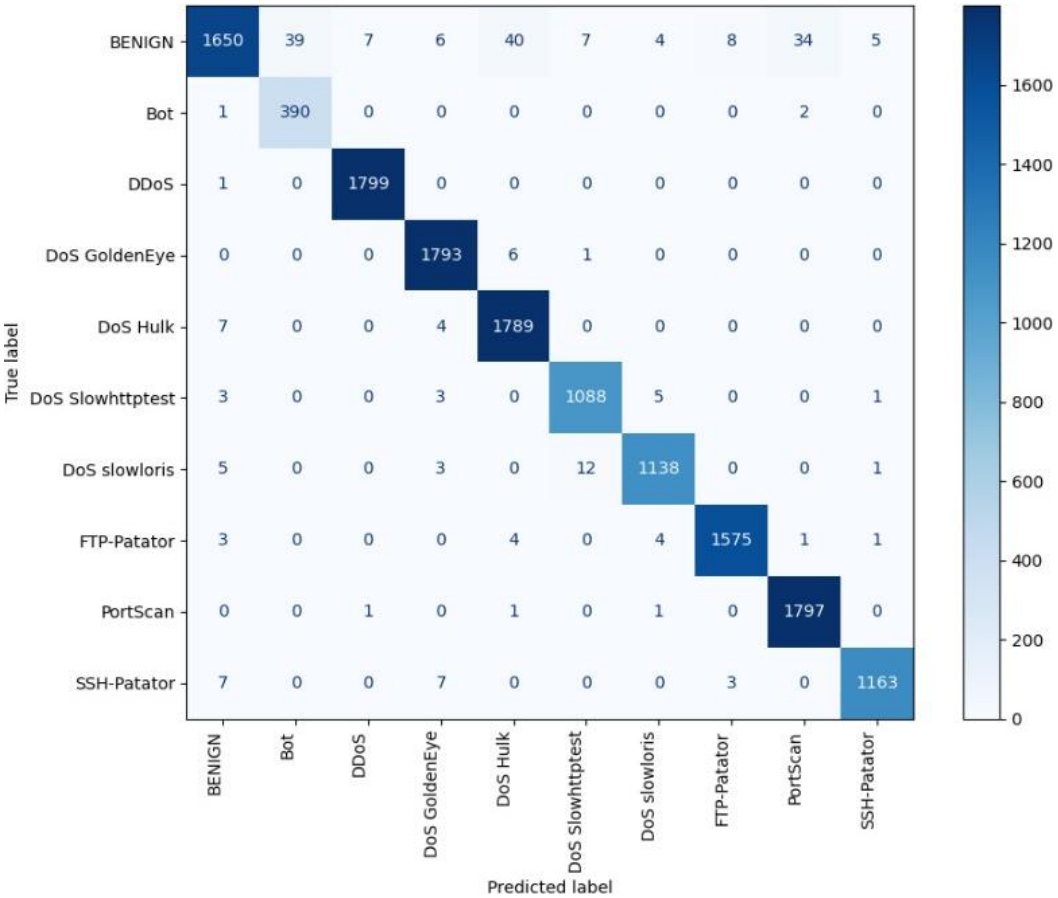

Figure 10. Multiclass confusion matrix for CICIDS2017 results

The proposed CNN–RBF–Attention model consistently achieves the best performance across all metrics in both binary and multiclass settings. By combining CNN-based feature extraction with localized RBF decision regions and attention-guided weighting, the hybrid model reduces false negatives while maintaining a low false positive rate. The improvement is particularly pronounced in multiclass classification, where attention-enhanced prototype weighting enables more precise separation of minority and overlapping attack categories. To avoid redundancy, only the confusion matrix and ROC curve of the proposed model are presented in Figures 10 and 11. The multiclass confusion matrix (Figure 10) exhibits strong diagonal dominance across all classes, indicating effective separation between benign traffic and multiple attack types, including low-frequency categories. Misclassifications are limited and primarily occur among behaviorally similar DoS variants, which are known to be difficult to distinguish in flow-based IDS datasets.

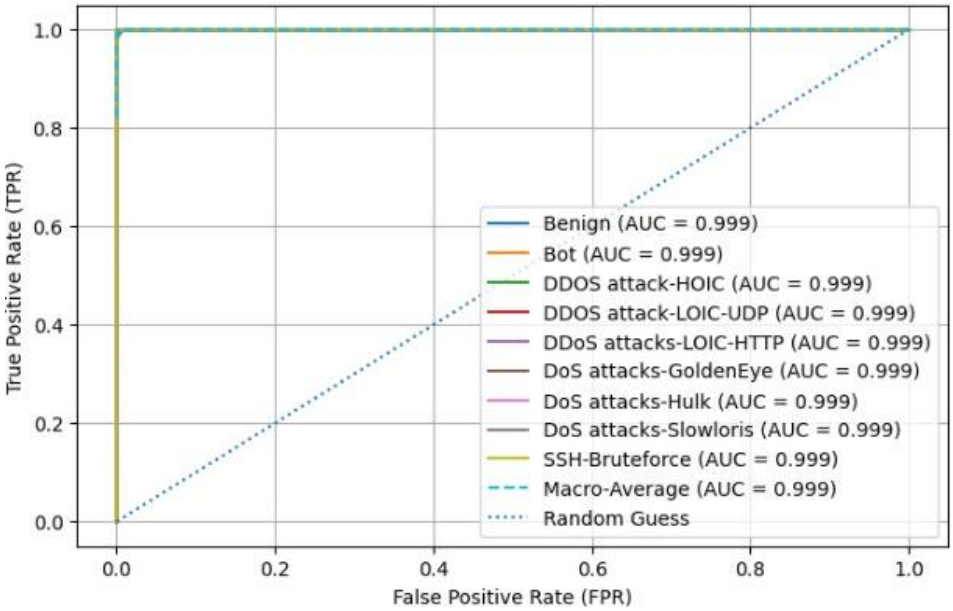

Figure 11. Multiclass ROC-AUC for CICIDS2017 results

The ROC curve in Figure 11 further confirms the robustness of the proposed model. While AUC values are near-perfect for most classes, the curve shows a steeper increase in the true positive rate at low false–positive rates, which is the most critical operating regime for real-world intrusion detection systems. This behavior demonstrates that the proposed model can detect rare and stealthy attacks while keeping false alarms at a manageable level for security operations. The results on CICIDS2017 validate the effectiveness of integrating deep hierarchical feature learning, localized prototype-based classification, and attention mechanisms. The proposed hybrid architecture demonstrates improved robustness under severe class imbalance. It provides more reliable detection of minority attacks, making it suitable for deployment in practical IDS environments where controlled false alarms and high recall are essential.

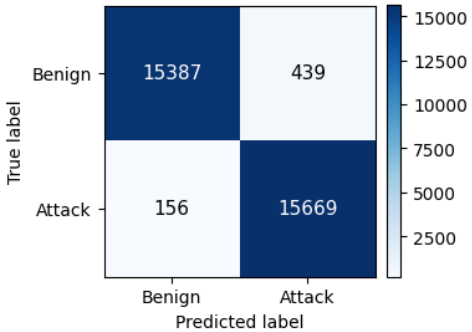

Figure 12. Binary class confusion matrix for CICIDS2017 results

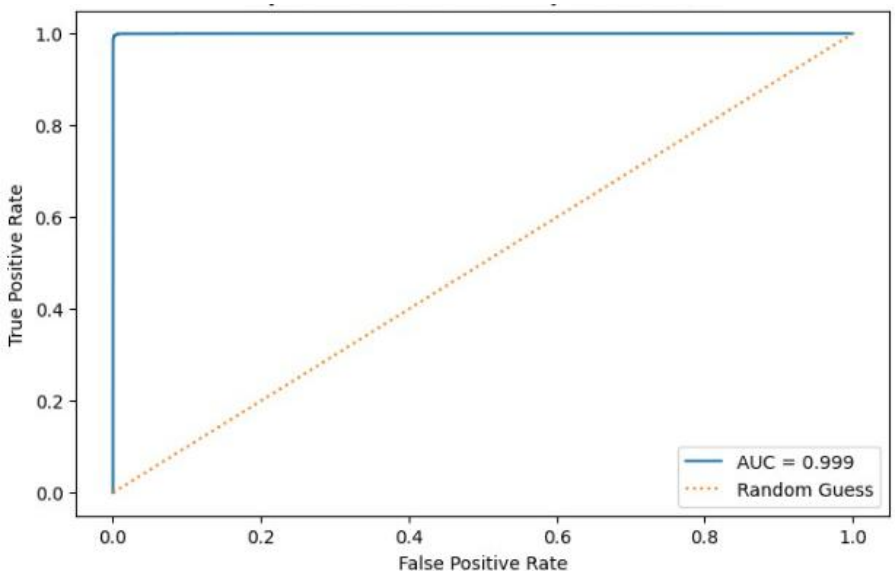

Figure 13. Binary class ROC-AUC for CICIDS2017 results

4.3 Comparison with Existing Intrusion Detection Methods

To assess the competitiveness of the proposed approach, Table 5 compares the hybrid CNN–RBF–Attention model with representative state-of-the-art intrusion detection methods reported in the literature. The comparison covers both binary and multiclass classification tasks on the CICIDS2017 and CICIDS2018 datasets, which are among the most widely used benchmarks for network intrusion detection.

Table 5. Comparison with existing intrusion detection methods on CICIDS2017 and CICIDS2018.

Models	Ref	Dataset	Classification Type	Acc. (%)	Prec. (%)	Rec. (%)	F1 (%)
CNN–LSTM	[14]	CICIDS2017	Multiclass	98.67	97.21	47.00	93.32
CNN–LSTM	[21]	CICIDS2017	Multiclass	96.21	NR	NR	NR
CNN–LSTM	[8]	CICIDS2018	Binary	98.80	98.50	98.60	98.60
CNN–RNN	[19]	CICIDS2018	Multiclass	98.80	98.64	99.15	99.03
DL–BiLSTM	[1]	CICIDS2018	Multiclass	98.61	66.52	73.18	67.23
Proposed method	Ours	CICIDS2017	Binary	98.12	98.12	98.12	98.12
			Multiclass	98.92	98.95	98.92	98.92
		CICIDS2018	Binary	99.81	99.81	99.81	99.81
			Multiclass	99.54	99.55	99.54	99.54

The proposed model achieves performance competitive with, and in several cases superior to, existing deep-learning-based IDS approaches, particularly in recall and F1-score. This improvement is most evident in multiclass scenarios, where severe class imbalance and overlapping attack patterns typically degrade the performance of conventional CNN, RNN, or LSTM-based models. While recurrent architectures such as CNN–LSTM and BiLSTM achieve strong accuracy on dominant classes, they often exhibit substantial recall degradation for minority attacks, as reflected in several prior works that report incomplete or imbalanced metric sets.

In contrast, the proposed hybrid architecture maintains consistently high recall across both datasets and classification settings. This behavior can be attributed to the combination of hierarchical CNN feature extraction, localized RBF prototype modeling, and attention-based weighting, which together enhance sensitivity to rare and low-frequency attack patterns without inflating false positives. Unlike recurrent or ensemble-based approaches, the proposed model achieves robustness with a relatively lightweight, interpretable architecture, which is advantageous for deployment in resource-constrained or real-time IDS environments. It is also notable that several prior studies report only accuracy, omitting recall and

F1-score, limiting their suitability for security-critical evaluation where false negatives carry high operational risk. The consistent reporting of recall, precision, and F1-score in this work enables a more reliable assessment of detection effectiveness and highlights the practical strengths of the proposed model in realistic intrusion detection scenarios.

4.4. Discussion

The results on both CICIDS2018 and CICIDS2017 confirm that combining CNN-based feature learning, RBF-based nonlinear mapping, and attention-based weighting leads to more reliable intrusion detection than standalone CNN or RBF models. The most consistent gains are observed in recall and F1-score, especially for minority and low-frequency attack classes, which are critical in realistic IDS settings. The RBF-only model struggles with overlapping and rare attack patterns, while the CNN improves performance through hierarchical representation learning but remains sensitive to class imbalance. The proposed hybrid architecture effectively bridges these limitations: the RBF layer creates localized decision regions in the CNN feature space, and the attention mechanism selectively amplifies the most discriminative prototypes. This interaction substantially reduces false negatives without increasing false positives.

Importantly, the hybrid model performs better in low false-positive-rate regions, which is essential for operational deployment, where excessive alerts are unacceptable. Compared to recurrent or ensemble-based IDS models, the proposed approach achieves competitive performance with lower architectural complexity and improved interpretability, making it well-suited for practical, real-time intrusion detection systems.

5. Conclusions

This study proposed an attention-enhanced hybrid CNN–RBF framework for network intrusion detection in highly imbalanced traffic environments. The model integrates three complementary components: hierarchical feature extraction through a CNN, localized nonlinear classification via RBF prototypes, and an attention mechanism that adaptively weights discriminative prototypes. This design directly addresses key limitations of existing IDS models, particularly the weak detection of minority and overlapping attack classes and the sensitivity of global classifiers to severe class imbalance.

Experimental results on the CICIDS2017 and CICIDS2018 benchmark datasets demonstrate that the proposed framework consistently outperforms standalone CNN and RBF baselines in both binary and multiclass settings. The most significant gains are observed in recall and F1-score, especially for low-frequency attack categories that are typically missed by conventional deep learning models. Confusion matrix and ROC analyses further confirm that the hybrid model operates reliably in low false-positive-rate regions, which is critical for practical deployment in security operations centers. These findings support the main research objective of improving detection sensitivity while maintaining controlled false alarms under realistic traffic imbalance.

From a methodological perspective, the results show that performance improvements are not driven solely by architectural complexity or accuracy, but by the synergy among deep feature learning, prototype-based local decision boundaries, and attention-guided refinement. This combination enables the model to focus on subtle and rare intrusion patterns while preserving interpretability and computational efficiency. Compared to recurrent or ensemble-heavy IDS architectures, the proposed framework achieves competitive or superior performance with a lighter and more modular design, making it suitable for real-time or resource-constrained environments. Despite these promising results, several limitations remain. First, the evaluation is limited to offline experiments on static benchmark datasets. Real-world deployment introduces additional challenges, such as concept drift, evolving attack strategies, and strict latency constraints, which this study does not address. Second, although SMOTE improves class balance during training, synthetic samples may not fully capture the complexity of adversarial traffic, potentially affecting generalization to unseen attack behaviors.

Future work will therefore focus on extending the proposed framework toward operational settings. Key directions include real-time evaluation and integration with existing NIDS platforms (e.g., Snort, Suricata, Zeek), the development of online or incremental learning strategies to handle concept drift and zero-day attacks, evaluation on more recent and diverse datasets, and the incorporation of explainability mechanisms such as attention visualization

and prototype relevance analysis to support security analysts better. Overall, the proposed CNN–RBF–Attention architecture provides a robust and effective foundation for intrusion detection in complex and imbalanced network environments. By unifying deep representation learning, prototype-based classification, and adaptive attention within a single framework, this work provides a practical, extensible approach toward more reliable, deployable intelligent intrusion detection systems.

Author Contributions: Conceptualization: F.K.; Methodology: F.K.; Software: F.K.; Validation: T.N.; Formal analysis: T.N.; Investigation: F.K.; Resources: F.K; Data curation: F.K.; Writing—original draft preparation: F.K; Writing—review and editing: F.K; Visualization: F.K; Supervision: T.N. All authors have read and agreed to the published version of the manuscript

Funding: This research received no external funding.

Data Availability Statement: The data used in this study are publicly available benchmark datasets for network intrusion detection. Specifically, the CICIDS2017 and CICIDS2018 datasets were obtained from the Canadian Institute for Cybersecurity (CIC), University of New Brunswick. These datasets are publicly accessible for research purposes at: https://www.unb.ca/cic/datasets/. No new datasets were generated during this study.

Conflicts of Interest: The authors declare no conflict of interest

References

[1] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment," *Sensors*, vol. 23, no. 13, p. 5941, Jun. 2023, doi: 10.3390/s23135941.

[2] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, "A Comprehensive Review of Cyber Security Vulnerabilities, Threats, Attacks, and Solutions," *Electronics*, vol. 12, no. 6, p. 1333, Mar. 2023, doi: 10.3390/electronics12061333.

[3] M. A. Rahman, G. A. Francia, and H. Shahriar, "Leveraging GANs for Synthetic Data Generation to Improve Intrusion Detection Systems," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 4, pp. 429–439, Feb. 2025, doi: 10.62411/faith.3048-3719-52.

[4] A. A. Hammad and F. T. Jasim, "Adaptive Cyber Defense using Advanced Deep Reinforcement Learning Algorithms: A Real-Time Comparative Analysis," *J. Comput. Theor. Appl.*, vol. 2, no. 4, pp. 523–535, Apr. 2025, doi: 10.62411/jcta.12560.

[5] F. Erlacher and F. Dressler, "On High-Speed Flow-Based Intrusion Detection Using Snort-Compatible Signatures," *IEEE Trans. Dependable Secur. Comput.*, vol. 19, no. 1, pp. 495–506, Jan. 2022, doi: 10.1109/TDSC.2020.2973992.

[6] Vasudev Karthik Ravindran, Sharad Shyam Ojha, and Arvind Kamboj, "A Comparative Analysis of Signature-Based and Anomaly-Based Intrusion Detection Systems," *Int. J. Latest Technol. Eng. Manag. Appl. Sci.*, vol. 14, no. 5, pp. 209–214, Jun. 2025, doi: 10.51583/IJLTEMAS.2025.140500026.

[7] M. B. Umair *et al.*, "A Network Intrusion Detection System Using Hybrid Multilayer Deep Learning Model," *Big Data*, vol. 12, no. 5, pp. 367–376, Oct. 2024, doi: 10.1089/big.2021.0268.

[8] G. Singh and M. Bansal, "Robust and Scalable Deep Learning Framework for Anomaly Detection in Large-Scale Network Security Systems," *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 17S, 2024, doi: 10.17762/ijisae.v12i17s.7685.

[9] M. A. Talukder *et al.*, "Machine learning-based network intrusion detection for big and imbalanced data using oversampling, stacking feature embedding and feature extraction," *J. Big Data*, vol. 11, no. 1, p. 33, Feb. 2024, doi: 10.1186/s40537-024-00886-w.

[10] A. Yulianto, P. Sukarno, and N. A. Suwastika, "Improving AdaBoost-based Intrusion Detection System (IDS) Performance on CIC IDS 2017 Dataset," *J. Phys. Conf. Ser.*, vol. 1192, p. 012018, Mar. 2019, doi: 10.1088/1742-6596/1192/1/012018.

[11] V. Hnamte and J. Hussain, "DCNNBiLSTM: An Efficient Hybrid Deep Learning-Based Intrusion Detection System," *Telemat. Informatics Reports*, vol. 10, p. 100053, Jun. 2023, doi: 10.1016/j.teler.2023.100053.

[12] Y. Xiao, C. Xing, T. Zhang, and Z. Zhao, "An Intrusion Detection Model Based on Feature Reduction and Convolutional Neural Networks," *IEEE Access*, vol. 7, pp. 42210–42219, 2019, doi: 10.1109/ACCESS.2019.2904620.

[13] C. Asuai *et al.*, "Enhancing DDoS Detection via 3ConFA Feature Fusion and 1D Convolutional Neural Networks," *J. Futur. Artif. Intell. Technol.*, vol. 2, no. 1, pp. 145–162, Jun. 2025, doi: 10.62411/faith.3048-3719-105.

[14] P. Sun *et al.*, "DL-IDS: Extracting Features Using CNN-LSTM Hybrid Network for Intrusion Detection System," *Secur. Commun. Networks*, vol. 2020, pp. 1–11, Aug. 2020, doi: 10.1155/2020/8890306.

[15] M. Amirian and F. Schwenker, "Radial Basis Function Networks for Convolutional Neural Networks to Learn Similarity Distance Metric and Improve Interpretability," *IEEE Access*, vol. 8, pp. 123087–123097, 2020, doi: 10.1109/ACCESS.2020.3007337.

[16] Z. Zhang, "Pattern Classification Based On Radial Basis Function Neural Network," in *2020 5th International Conference on Smart Grid and Electrical Automation (ICSGEA)*, Jun. 2020, pp. 213–216. doi: 10.1109/ICSGEA51094.2020.00052.

[17] T. Li and J. Qiao, "A novel radial basis function neural network classifier based on three-way decisions," *Eng. Appl. Artif. Intell.*, vol. 141, p. 109811, Feb. 2025, doi: 10.1016/j.engappai.2024.109811.

[18] A. Çetin and S. Öztürk, "Comprehensive Exploration of Ensemble Machine Learning Techniques for IoT Cybersecurity Across Multi-Class and Binary Classification Tasks," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 4, pp. 371–384, Feb. 2025, doi: 10.62411/faith.3048-3719-51.

[19] E. U. H. Qazi, M. H. Faheem, and T. Zia, "HDLNIDS: Hybrid Deep-Learning-Based Network Intrusion Detection System," *Appl. Sci.*, vol. 13, no. 8, p. 4921, Apr. 2023, doi: 10.3390/app13084921.
[20] W. Zhao and Z. Zhao, "Providing a hybrid approach to increase the accuracy of intrusion detection systems in computer networks," *J. Eng. Appl. Sci.*, vol. 71, no. 1, p. 123, Dec. 2024, doi: 10.1186/s44147-024-00404-y.
[21] M. Sajid *et al.*, "Enhancing intrusion detection: a hybrid machine and deep learning approach," *J. Cloud Comput.*, vol. 13, no. 1, p. 123, Jul. 2024, doi: 10.1186/s13677-024-00685-x.
[22] J. P. Ntayagabiri, Y. Bentaleb, J. Ndikumagenge, and H. El Makhtoum, "OMIC: A Bagging-Based Ensemble Learning Framework for Large-Scale IoT Intrusion Detection," *J. Futur. Artif. Intell. Technol.*, vol. 1, no. 4, pp. 401–416, Feb. 2025, doi: 10.62411/faith.3048-3719-63.
[23] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, 2018, pp. 108–116. doi: 10.5220/0006639801080116.
[24] T. T. T. Nguyen and G. Armitage, "A survey of techniques for internet traffic classification using machine learning," *IEEE Commun. Surv. Tutorials*, vol. 10, no. 4, pp. 56–76, 2008, doi: 10.1109/SURV.2008.080406.
[25] Z. Wang, Y. Zeng, Y. Liu, and D. Li, "Deep Belief Network Integrating Improved Kernel-Based Extreme Learning Machine for Network Intrusion Detection," *IEEE Access*, vol. 9, pp. 16062–16091, 2021, doi: 10.1109/ACCESS.2021.3051074.
[26] A. Balla, M. H. Habaebi, E. A. A. Elsheikh, M. R. Islam, and F. M. Suliman, "The Effect of Dataset Imbalance on the Performance of SCADA Intrusion Detection Systems," *Sensors*, vol. 23, no. 2, p. 758, Jan. 2023, doi: 10.3390/s23020758.
[27] G. Sameera, R. V. Vardhan, and K. V. S. Sarma, "Binary classification using multivariate receiver operating characteristic curve for continuous data," *J. Biopharm. Stat.*, vol. 26, no. 3, pp. 421–431, May 2016, doi: 10.1080/10543406.2015.1052479.
[28] I. A. Vergara, T. Norambuena, E. Ferrada, A. W. Slater, and F. Melo, "StAR: a simple tool for the statistical comparison of ROC curves," *BMC Bioinformatics*, vol. 9, no. 1, p. 265, Dec. 2008, doi: 10.1186/1471-2105-9-265.