Feature-Level Fusion in Hybrid CNN-BiLSTM Deep Learning Architecture for Multi-Source Intrusion Detection Systems

Ram Naresh Sharma[1*], Jitendra Singh Kushwah[2] and Pritaj Yadav[3]

[1,3*]Department of Computer Science and Engineering, Rabindranath Tagore University, Bhopal, India.

[2]Department of Information Technology, Institute of Technology and Management, Gwalior, India.

Abstract

The rapid evolution of cyberattacks and the heterogeneity of modern network environments demand intrusion detection systems (IDS) capable of generalizing across diverse traffic distributions. This study proposes a hybrid dual-branch deep learning architecture that integrates CNN- and BiLSTM-based feature extractors with an attention mechanism, specifically designed for fused multi-source data. A unified dataset was constructed by combining CICIDS-2017 and UNSW-NB15, resolving schema mismatches and harmonizing feature semantics to form a comprehensive representation of contemporary attack behaviors. The fusion process enriched the statistical temporal characteristics of network flows, enabling the model to learn invariant signatures across distinct attack families. Experimental results show strong performance across all evaluation phases. The model achieved 96.80% training accuracy and 96.77% validation accuracy, demonstrating excellent convergence and minimal overfitting. On unseen traffic, the architecture achieved 89.88% accuracy and a macro-F1 of 0.8990, indicating robust generalization across mixed distributions. Class-wise analysis revealed high reliability for complex attack types such as DoS (0.93 F1), Backdoor (0.91 F1), and Reconnaissance (0.90 F1), while minor challenges arose with benign reconnaissance overlap. Threshold optimization further stabilized predictions for minority classes. The model outperforms conventional single-branch

CNN/LSTM IDS frameworks by exploiting complementary spatial temporal features and emphasizing discriminative patterns through attention. Overall, the proposed fusion-driven hybrid architecture provides a scalable and accurate solution for unified intrusion detection across heterogeneous network domains. The results confirm that multi-source feature fusion, combined with a parallel deep learning pipeline, significantly enhances adaptability, robustness, and detection capabilities in real-world network security environments.

Keywords: Intrusion Detection System (IDS), Deep Learning, Machine Learning, Feature-Level Fusion, Industrial Cybernetics

1 Introduction

As networked systems increasingly appear and grow in complexity, as much as the scale of the Industrial Internet of Things (IIoT), cloud networks, and critical infrastructures continue to rise, the issue of cybersecurity becomes a priority[1], [2]. The use of traditional security mechanisms like firewalls and antivirus systems are not enough to identify the complex cyberattacks and to implement the Intrusion Detection Systems (IDS) to be constantly monitored and to perform the proactive mitigation of threats [3], [4]. IDS are a critical defense line since they examine network traffic, system logs and device behavior to determine malicious traffic in real time[1], [5]. They offer early warning messages, improve forensic analysis and robustness in heterogeneous and large-scale industrial settings [6], [7]. Over the years, IDS have undergone a steep evolution as they have progressed to more complex machine learning (ML) and deep learning (DL)-based systems[8]. IDS Signature-based IDS identify attacks by comparing network traffic to known attack signatures, which is highly accurate when identifying known attacks but has weak ability against zero-day or polymorphic attacks[9]. In comparison, anomaly based IDS detects outliers of normal systems operation on the basis of statistical or heuristic models, thus allowing previously unknown attacks to be captured, but there is usually a high false-positive rate and model drift [10]. Specification-based IDS are a hybrid between signature-based and anomaly-based, and they impose predefined behavioral constraints on systems, and are especially appropriate in environments with critical infrastructure, such as smart grids and industrial control systems (ICS) [11].

ML and DL methods have greatly augmented the performance of IDS that enables dynamic and automatic recognition of sophisticated attack patterns [12]. Decision trees, random forests, and ensemble models are machine learning algorithms that have been implemented effectively to identify intrusion in IIoT and industrial settings [3], [13], [14]. ML-based IDS enhance the detection rates but are highly reliant on powerful feature extractors and classifiers because network traffic data is frequently both high-dimensional, non-homogenous, and disproportionate [15], [16]. The IDS can be improved in terms of its discriminative power by employing feature-based techniques that allow distinguishing normal and suspicious behaviors. Neural network models can be improved further by the fact that neural networks learn hierarchical representations in raw network traffic automatically [17]-[19]. CNNs are especially well-suited to identify the space-related patterns in the network data, which enables one to identify the regular pattern of the attack signatures [19], [20]. The temporal dependencies are represented by Recurrent Neural Networks (RNN) such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), which can be used to identify sequential and multi-step attacks that change with time [15], [21]23]. Such hybrid architectures as CNN-LSTM are based on spatial and temporal analysis to obtain high detection rates and low false-positive [12].

Generative methods have also been used, such as Generative Adversarial Networks (GAN), to augment learning data, solve class imbalance, and enhance the resilience of IDS [24], [25]. Equally, the models based on transfer learning, like TL-CNN-IDS, enable the application of knowledge in one network setting to be transferred to a different network setting, eliminating retraining processes and enhancing detection in novel deployment conditions [26], [27]. They have been particularly useful in IIoT, SDN and cloud network settings, where the problems of heterogeneity and scalability are currently salient[1], [28], [29]. Multi-feature and data fusion technique application is another important development in the field of IDS. The idea behind data fusion is to combine information provided by different sources to increase the accuracy of detection and decrease false positives and increase resilience [13]. At the feature-level, fusion is used to produce highly discriminatory features by merging features of multiple datasets to enhance the performance of ML/DL models[13]. Sensor-level fusion fuses raw information across two or more monitoring locations to provide coverage and contextual understanding in sophisticated industrial environments and induces synchronization issues and computational costs [6]. IDS reliability is further enhanced by decision-level fusion where the results of the various models are combined by means of majority voting or probabilistic inference [30].

The growing use of IDS in ICS, IIoT, and clouds environments highlights the necessity of real-time, adaptive, and hybrid detection schemes,[6], [17], [28]. ML and DL-based IDS has the ability to identify the known and unknown threat, minimize the false alarm and can be deployed in large network [1], [12]. Furthermore, multi-feature fusion guarantees the ability to use the heterogeneous data sets effectively and can detect the threats better and offer strong cybersecurity protection [13], [30]. Irrespective of these improvements, there are still a number of challenges. IDS have to deal with high-dimensional data, changing attack vectors, and operational complexity of deep learning models[8], [15]. Although multiple data sources increase the accuracy, they need selection of features and mechanisms of synchronization so that the redundancy of information is not acquired and the process becomes efficient. Moreover, adversarial attacks, zero-day attacks, and insider threats remain a major threat and research is still needed on more resilient, adaptive, and smart IDS frameworks [31]33]. It is possible to say that the integration of the ML and the DL approach and the feature-based and multi-source fusion of the data led to the appearance of the functions of the IDS in the context of the modern network to a significant extent [1], [13], [30]. CNNs, RNNs, hybrid, and GANs enable it to detect the spatial, temporal, and complex attack patterns appropriately [17], [20], [24]-[26]. Moreover, feature-level and sensor-level fusion system may enhance the credibility of discovery and provide viable solutions to IIoT, ICS, and cloud-based network [6], [13], [30]. Despite this, high-dimensional data, complex model and dynamic cyber threats research issues remains unsolved, and thus, it is a good news that more research on scalable, adaptive, and intelligent IDS architectures capable of meeting the demands of the future smart industrial and cyber-physical world is still being researched [8], [31], [32].

2 Literature Review

Intrusion Detection Systems (IDSs) have become an essential security component of the present-day networked ecosystem, particularly as cyberattacks are growing in size and complexity. The conventional IDS designs can no longer withstand the changes in the dynamic threat environment and are required to incorporate machine learning (ML), deep learning (DL), and hybrid analytics. The initial background reviews indicate the relevance of anomaly-based IDS (AIDS) and provide detailed taxonomies of detection techniques, attack types, datasets and assessment issues [38]. On the same note, benchmarking studies have pointed out inconsistency in the choice of algorithms, use of outdated datasets, and superficial validation methods and demanded more organized evaluation methods [37].

Comparative studies of both ML and DL models indicate evident performance improvements using deep neural networks and feature-learning methods. Rawat et al. have shown that the use of integrated unsupervised feature extraction and DNNs shows great results compared to classical ML techniques when evaluated on NSL-KDD and SDN settings [34]. This is further enhanced by hybrid feature selection and ensemble learning in which feature reduction and clustering with no supervision provide significant improvements in detection accuracy at a lower computational cost [35]. Other models optimized by ANNs show similar footing, with the hyperparameter search improving the performance of the models on NSL-KDD and CICIDS2017 with better than 99 percent accuracy [50].

The deep learning methods remain popular in the area of the detection of zero-day attacks and the increase of the robustness of the models. DL-based IDS systems that are developed based on UNSW-NB15 are more flexible and more accurate in their classification capabilities, particularly when they are developed with multi-layered architectures to learn complex behavioral patterns [40]. Nonetheless, machine learning-based IDSs can be easily affected by adversarial attacks, which is a widespread security risk. The use of evolutionary computation and GAN-generated adversarial examples has demonstrated that ML-based IDSs have serious flaws, which highlights the necessity to create model enhancements and design more resilient and robust models [42].

In order to improve the work of IDS, various studies have addressed various ML classifiers. SVM, DT, RF, and DJ models are fully tested on CICIDS2017 with SVM demonstrating high accuracy, precision, and recall in most cases [39]. Simultaneously, hybrid DL systems based on convolutional and recurrent networks (CNN + RNN) have been shown to perform better in local and temporal feature extraction of traffic flows. It is important to note that a hybrid CRNN-based IDS scored 98.90 percent on

CICIDS2018, which indicates that deep hybrid IDS are prevailing in the literature [45]. There are also new problems created by cloud-based security in which false alarms are high, thus making operations untrustworthy. A hybrid Cu-LSTMGRU architecture using Pearson correlation-based feature selection is more effective and efficient in cloud IDS [44].

Further, biometric security research enhances the development of IDS, as it proposes privacy-saving authentication systems. Multimodal biometrics is secured by application of post-quantum cryptography and homomorphism encryption that ensures long-term data protection [51]. Feature level fusion with trained ML classifiers has been effective in improving multimodal biometric authentication, is even more resilient to variations in the environment [49].

Spatiotemporal traffic modeling via graph convolutional networks and bidirectional GRUs is also discussed in recent studies to capture the high-dimensional and correlated network behavior, which have potential benefits in the next-generation IDS research [47].

According to the literature review, there are a number of research gaps in the development of the IDS today. Although the methods of ML and DL have enhanced the accuracy of detection, most models are still incapable of imbalanced and high-dimensional data, restraining their extrapolation with different network settings [47]. Also, the adversarial attacks disclose the weaknesses of the existing ML-based IDSs, which underscores the necessity of resolute and resilient frameworks [42]. Multi-source data fusion and hybrid architectures are still understudied, especially in cloud, IoT and critical infrastructure applications [26],[46],[52-57]. Further exploration is also needed on efficiency in deployment in real-time and optimization of computational efficiency.

3 Proposed Research Methodology

The suggested study presents a new multi source fusion based intrusion detection model established on a hybrid neural structure of convolutional, recurrent, and attention networks. The current research, in contrast to the previous studies that use a single dataset or a traditional set of features, adds a cross-domain harmonized dataset of intrusions, which is formed by combining CICIDS-2017 and UNSW-NB15 with the help of an extended feature alignment and semantic mapping pipeline. In addition, the methodology proposes domain-based feature construction and dual-branch deep learning framework that collectively learns aggregated and temporal features related to traffic. In this section, every step of the suggested strategy is formalized.

The suggested piece contains a full cross-dataset semantic feature fusion framework which balances the heterogeneous network flow features into a single representation, and thus, allows the effortless incorporation of fundamentally varied datasets of IDS. A new engineered behavior-sensitive set is created to increase the discriminatory power, which includes directionality ratios, flow-intensity measures, congestion measures, and a list of log-transformed statistical measures, which are not present in raw data sources. The paper also introduces a hybrid SMOTE-ENN and de-duplication based data balancing system, the first IDS method to implement a three-layer leak-prevention system and integrates oversampling with non-overlapping and contamination-free training, validation, and testing partitions. Based on the enhanced characteristics, a new dual-branch hybrid deep learning framework is proposed with one branch using Conv1D, BiLSTM, and an Attention mechanism to model the temporal sequences and the other branch learning high-level semantic patterns using dense transformations of aggregated static features. This architectural merger in itself contributes a lot to generalization in various traffic behaviors. Lastly, the systematic harmonization and dataset fusion develop a more generalizable IDS dataset that contains both old and new attack patterns, providing a more reliable alternative to the current single-source IDS benchmarks, and enhancing practicality in the real world.

3.1 Dataset Description and Rationale for Fusion

The traffic behavior, type of attack, and the philosophy of feature engineering of CICIDS-2017 and UNSW-NB15 differ radically. CICIDS-2017 allows capturing real-world high volume attack scenarios under realistic conditions, whereas UNSW-NB15 contains systematic low-level intrusions, which are conducted in a controlled environment of a cyber-range. Combination of these sets helps to address the bias of the dataset, increases the diversity of attacks and reduces overfitting to particular behavior patterns in the environment one of the weaknesses of current research on IDS. In this way, the combination creates a more detailed cyber-threat environment, enhancing the capability of the downstream model to identify real-life multi-modal attacks Because of the severe imbalance in CICIDS-2017 in which benign traffic is the most active with more than 2.27 million flows, downsampling was carried out to maintain meaningful distribution with minority attack samples. Fine-grained labels were grouped into six broader categories to fit UNSW-NB15 taxonomy. CICIDS-2017 had 907,646 harmonized sample inputs after preprocessing. UNSW-NB15 also had overlapping and very rare categories of attack that had the potential of skewing the learning. The six categories were obtained after the removal of non-informative classes and the consolidation of the rare ones into the same category as CICIDS-2017. The alignment facilitated successful feature fusion and homogenous downstream classification.

3.2 Feature Alignment and Semantic Harmonization

A new semantic mapping pipeline was used in order to map both datasets onto the same feature space. CICIDS attributes were mapped to UNSW counterparts according to the duration of a flow, the volume of a flow, directionality, and semantics of packets. Further normalization was done to ensure that both the datasets worked on the same units (e.g. duration was changed to seconds). This led to a final schema with ten numeric features and one categorical labelling which could be directly concatenated without structural inconsistency. Both the harmonized datasets were combined into a single dataset. The merging presented compatible network behaviors CICIDS brought in volumetric attacks in modern times, and UNSW stealth attacks such as reconnaissance attacks and exploitation. This combination is one of the primary novelties, and it provides a generalizable intrusion dataset with less bias and greater variety.

In order to build a coherent, cross-dataset intrusion detection system, a stringent feature matching and semantic balancing measure was adopted. The main goal was to homogenize heterogeneous network flow features generated by two radically different IDS data CICIDS and UNSW-NB15 so that both data sets can play significant roles in the combined data set. It was done by renaming, reorganization, cleaning, and statistical harmonization of feature distributions.

The first approach was the choice of a stable sub-set of flow-level attributes that are semantically equivalent. CICIDS and UNSW have different naming conventions and measurement formats; therefore, the CICIDs data was directly renamed to be in the schema of the UNSW. The characteristics in the form of Flow Duration, Forward Packets, Backward Bytes, Average Segment Sizes, and TCP window were transformed into a single form: dur, spkts, dpkts, s bytes, dbytes, rate, smean, dmean, swin, dwin. This standardization guarantees that every feature has the same operational semantics across datasets, and it does not have ambiguity and can be easily compared.

This was followed by a uniform column format by simply taking the harmonized set of features out of each dataset. Based on the common list of columns, both of the datasets were cut with the same schemas and then joined together to create the final fused datasets. This promoted structural homogeneity, where ensuing deep learning models are enabled to handle inputs without biases ascertained by the data.

The step of detailed statistical harmonization was then followed, in which the distributions of every common feature were compared by histograms. The 1 st 99 th percentile range was used to remove extreme outliers and the invalid values (NaN, Inf) were cleaned. This was an essential step, because CICIDS and UNSW are very different in terms of data generation methodology resulting in incompatible numeric scales, skewness and density distributions. Through the rectification of these discrepancies, fused dataset will not only achieve cross-dataset stability but also reduce domain shift and increase generalization of intrusion detection models.

The derivation formalizes the preprocessing pipeline of any one feature by initially eliminating invalid values like infinities and NaNs, and then narrowing down the rest of the samples to the range between the 1st percentile and 99th percentile. This is mathematically equivalent to cleaning and outlier trimming and makes sure that all datasets have similar distributions of noise-free features to be fused and model trained.

$$\hat{X}_d^{(i)} = \{\, x \in X_d^{(i)} \mid x \neq \infty,\ x \neq -\infty,\ x \neq \mathrm{NaN} \,\}, \tag{1}$$

$$\tilde{X}_d^{(i)} = \left\{ x \in \hat{X}_d^{(i)} \;\middle|\; P_1\left(X_d^{(i)}\right) \leq x \leq P_{99}\left(X_d^{(i)}\right) \right\}, \tag{2}$$

where,

$$\begin{aligned}
\hat{X}_d^{(i)} &= X_d^{(i)} \setminus \{\infty, -\infty, \mathrm{NaN}\},\\
P_1\left(X_d^{(i)}\right) &= \text{1st percentile of the feature } X_d^{(i)},\\
P_{99}\left(X_d^{(i)}\right) &= \text{99th percentile of the feature } X_d^{(i)},\\
\tilde{X}_d^{(i)} &= \text{Outlier-trimmed feature ensuring } P_1\left(X_d^{(i)}\right) \leq x \leq P_{99}\left(X_d^{(i)}\right).
\end{aligned} \tag{3}$$

Table 1 Aligned Common Features Across CICIDS and UNSW

Unified Feature	**CICIDS Original Name**	**UNSW Original Name**
dur	Flow Duration	dur
spkts	Total Fwd Packets	spkts
dpkts	Total Backward Packets	dpkts
sbytes	Total Length of Fwd Packets	sbytes
dbytes	Total Length of Bwd Packets	dbytes
rate	Flow Packets/s	rate
smean	Avg Fwd Segment Size	smean
dmean	Avg Bwd Segment Size	dmean
swin	Init_Win_bytes_forward	swin
dwin	Init_Win_bytes_backward	dwin
attack_cat	Label	attack_cat

Table 2 Semantic Fusion Layer: Combined Feature Interpretation

Unified Feature	**Semantic Meaning**	**Usage in Fusion**
dur	Total duration of flow	Baseline temporal intensity
spkts / dpkts	Directional packet volume	Behavior asymmetry
sbytes / dbytes	Payload distribution	Traffic burst analysis
rate	Packet emission speed	Attack activity frequency
smean / dmean	Segment size patterns	Congestion or flow structure
swin / dwin	TCP window behavior	Congestion anomaly cues

Table 3 CICIDS Dataset Statistical Summary

Feature	**Mean**	**Std**	**Min**	**Max**
dur	22.326443	39.04967	-0.000012	120.000000
spkts	6.543184	514.033900	1.000000	218658.000000
dpkts	6.609115	687.080800	0.000000	291260.000000
sbytes	363.996801	6529.805000	0.000000	2866110.000000
dbytes	10229.052435	1551715.000000	0.000000	641001400.000000
rate	84029.867081	284271.900000	-2000000.000000	3000000.000000
smean	40.833114	138.389400	0.000000	5940.857000
dmean	613.383646	912.714000	0.000000	5800.500000
swin	7221.727626	13042.550000	-1.000000	65535.000000
dwin	1118.640460	6183.855000	-1.000000	65535.000000

Table 3 summarizes all statistics of important network-flow attributes in the CICIDS dataset in a statistical form that includes the mean, deviation and range of values of these networks-flow attributes. It shows huge deviation in the features of bytes, packets, and rate, which serves as evidence of highly dynamic traffic behavior and existence of heavy-tailed patterns applicable in intrusion detection.

Table 4 UNSW-NB15 Dataset Statistical Summary

Feature	Mean	Std	Min	Max
dur	1.304573	5.348192	0.0	59.999999
spkts	23.710268	153.032698	1.0	10646.000000
dpkts	22.625100	130.694400	0.0	11018.000000
sbytes	10372.870061	196537.376456	24.0	14355770.000000
dbytes	16903.631734	171476.804520	0.0	14657530.000000
rate	47316.274710	119889.988299	0.0	1000000.000000
smean	162.622921	231.650628	24.0	1504.000000
dmean	149.395747	266.481097	0.0	1500.000000
swin	172.921642	119.129592	0.0	255.000000
dwin	166.097635	121.508690	0.0	255.000000

The metrics in Table 4 describe the statistics of vital traffic properties in the UNSW-NB15 dataset. As depicted, the values of attributes like bytes, packets, and rates are highly diverse, that is, the network activity is heterogeneous. The statistics help in normalization of data and create strong feature engineering of IDS models.

3.3 Data Preprocessing Framework

The suggested preprocessing framework proposes a set of strict and new pipeline that would generate a clean, behavior-rich, balanced, and contamination-free dataset that would be used in the high-fidelity intrusion detection research. It consists of a long stage of data cleaning and outlier removal, during which invalid or missing values are imputed based on median statistics to handle the weaknesses of having skewed or non-Gaussian distributions of network traffic. Percentile based thresholds are used in outlier removal to remove out of the ordinary noise, without affecting legitimate attack variation. This gives a stable and steady dataset that can be predictive of model learning especially when using deep learning designs, which are susceptible to abnormalities in the distribution of inputs.

Based on this, the framework includes behavior-based feature engineering, which adds a number of new traffic descriptors which cannot be found in raw datasets. These are directional byte and packet ratios, which quantify source destination asymmetry, flow intensity indicators which quantify burstiness, TCP window dynamics which are used to analyze congestion behavior, and zero-flow anomaly flags which are used to identify suspicious inactivity patterns. Further, heavy-tailed variables are also transformed to a log-form to standardize interest of a distribution and increase models sensitivity to subtle differences. These artificial capabilities form a significant contribution of the study and greatly enhance the capabilities of the dataset to detect subtle attempts of an intrusion that are typically not detected by standard feature sets. Derivation is formalizing the idea of feature engineering, where ratios, differences, intensity, and window-based measures are characterized to model bidirectional flow behavior. Zero-value flags are used to identify abnormal traffic patterns and log-transforms will address skewed distributions. Such mathematically constructed aspects complement the information, enhancing semantic coverage and raising the accuracy of intrusion detection of various attack types.

Ratio-based Features:

$$\text{pkt\_ratio} = \frac{s_{pkts}}{d_{pkts}+1}, \quad \text{byte\_ratio} = \frac{s_{bytes}}{d_{bytes}+1}, \quad \text{size\_ratio} = \frac{s_{mean}}{d_{mean}+1} \tag{4}$$

Difference Features:

$$\text{pkt\_diff} = |s_{pkts} - d_{pkts}|, \quad \text{byte\_diff} = |s_{bytes} - d_{bytes}|, \quad \text{size\_diff} = |s_{mean} - d_{mean}| \tag{5}$$

Flow Intensity Measures:

$$\text{total\_pkts} = s_{pkts} + d_{pkts}, \tag{6}$$

$$\text{total\_bytes} = s_{bytes} + d_{bytes}, \tag{7}$$

$$\text{bytes\_per\_pkt} = \frac{\text{total\_bytes}}{\text{total\_pkts}+1}, \tag{8}$$

$$\text{pkts\_per\_sec} = \frac{\text{total\_pkts}}{\text{dur}+10^{-6}} \tag{9}$$

Window-based Features:

$$\text{win\_ratio} = \frac{s_{win}}{d_{win}+1}, \quad \text{win\_diff} = |s_{win} - d_{win}| \tag{10}$$

Zero-value Flags:

$$\text{win\_zero\_flag} = \begin{cases} 1, & \text{if } s_{win} = 0 \ \lor \ d_{win} = 0 \\ 0, & \text{otherwise} \end{cases} \tag{11}$$

$$\text{zero\_pkt\_flag} = \begin{cases} 1, & \text{if } s_{pkts} = 0 \ \lor \ d_{pkts} = 0 \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

$$\text{zero\_byte\_flag} = \begin{cases} 1, & \text{if } s_{bytes} = 0 \ \lor \ d_{bytes} = 0 \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

Log-Transformed Features:

$$\log_*(x) = \ln(1+x), \quad \forall x \in \{\text{dur}, s_{bytes}, d_{bytes}, \text{rate}\} \tag{14}$$

In order to further narrow-down the dataset, correlation-based pruning is used to remove redundancy so that all features with correlation coefficients of +0.85 or greater are removed. This procedure minimizes multicollinearity, increases interpretability and minimizes the chances of overfitting so that the learning model concentrates on distinctive and informative features. This is then followed by the hybrid SMOTE-ENN approach of class balancing that oversamples the minority classes and undersamples

Edited Nearest Neighbors. This two-pronged approach increases the representation of minority classes without at the same time adding any noisy samples or samples that are on the borderline that can damage the stability of the classifier. The outcome is a balanced dataset which dramatically increases the G1-score of classes of attacks which were previously poorly represented.

This derivation is used to formalize the process of redundancy detection, where the features that have a zero variance are identified and the highly correlated features above a threshold. The upper triangle of the correlation removes redundant predictors to guarantee that noise and multicollinearity are removed. This increases the stability of the model, minimises overfitting, and increases interpretability prior to fusion-based intrusion detection modelling.

Let the dataset be represented as $X = \{x_1, x_2, \ldots, x_n\}$ with target variable $y =$ attack_cat. All non-target features form the matrix X_{num}.

1. Low-Variance Feature Detection:

$$\mathcal{L} = \left\{ f_i \;\middle|\; |\text{unique}(f_i)| \leq 1 \right\} \tag{15}$$

2. Correlation Matrix Computation:

$$C = [\rho(f_i, f_j)] = \left[\frac{\text{Cov}(f_i, f_j)}{\sigma_{f_i}\sigma_{f_j}} \right] \tag{16}$$

3. Upper-Triangle Search for Highly Correlated Features:

$$U = C \odot \mathbf{1}_{i<j} \tag{17}$$

where $\mathbf{1}_{i<j}$ is an indicator mask keeping only entries with $i < j$.

A feature is flagged as redundant if:

$$\mathcal{H} = \left\{ f_j \;\middle|\; \exists i : U_{ij} > \tau \right\}, \qquad \tau = 0.85 \tag{18}$$

Final Redundancy Set:

$$\mathcal{R} = \mathcal{L} \cup \mathcal{H} \tag{19}$$

Lastly, the framework has stringent splitting of datasets and data leakage. Three stage deduplication operation guarantees zero overlap on training, validation and test partitions this is a rare quality to performance standards in IDS research, as duplication based on flows may inflate performance statistics unconsciously. Such stringent leakage control is rarely used in other existing studies and, therefore, this methodological approach is a point to note. Together, the suggested preprocessing pipeline forms a very reliable and feature-rich and neutral data base that improves the overall performance of the models and allowed strong and reproducible research of IDS.

3.4 Proposed Hybrid Deep Learning Model Architecture

The proposed intrusion detection model comes up with a strong and creative hybrid deep learning network that combines convolutional, recurrent and attention-based networks and densely connected neural networks. This is the framework that is specifically tailored to use the enriched fused IDS dataset itself and efficiently tackle the limitations of the heterogeneous traffic behavior, class imbalance, and time and aggregated feature learning requirements. The model combines two complementary branches; one that emphasizes on sequential patterns and the other one concentrating on the static aggregated relationships, making it have a comprehensive insight on network flow traits that would make it better in detecting various attack vectors.

3.4.1 Input Representation and Feature Scaling

The input features are z-scaled by using StandardScaler to achieve model stability and successful convergence. This scaling removes inconsistencies in value ranges and avoids the influence of dominant values of high magnitude attributes. The data is then rearranged to a three dimensional tensor of the form (batch size, timesteps=1, features) to allow subsequent use with Conv1D and LSTM layers to be used. This reorganization is essential as it maintains the alignment of features, even though convolutional and sequential learning models can be used to locate local and contextual relationships in the input vector.

Let a mini-batch input be denoted by

$$\mathbf{X} \in \mathbb{R}^{B \times T \times F},$$

where B is the batch size, T the number of timesteps and F the number of features per timestep. In the implemented model $T = 1$ (single-step flows), but we keep T general for derivation.

Each feature is standardized (z-score) so that the network input is

$$\bar{\mathbf{X}} = \text{Standardize}(\mathbf{X}).$$

The network input layer passes $\bar{\mathbf{X}}$ forward.

3.4.2 Dual-Branch Hybrid Design

The architecture introduces a two-branch design that extracts complementary patterns from the input data.

Branch 1: Conv1D + BiLSTM + Attention (Temporal Feature Learning)

It is a branch that elicits sequential dependencies and time correlation of network flows. Conv1D layer consisting of 64 filters and a 1x1 kernel detects the local interactions between features. MaxPooling1D is used to stabilize representation but not to make an important dimensionality reduction. The obtained features are then inputted into a 128-unit Bidirectional LSTM which processes the sequence forward and backward to acquire more detailed contextual dependencies. There is then a custom attention mechanism, where attention weights are given to timesteps that are important, and the model is allowed to attend to behaviorally relevant flow attributes. Later, 128 and 64-unit dense layers fine-tune the learned representation and a dropout rate of 0.3 decreases overfitting and improves generalization.

Conv1D (kernel size $k = 1$, $N_c = 64$ filters).

Conv1D with kernel size 1 acts as an affine transform across features at each timestep:

$$\mathbf{C}_t = \phi\big(\mathbf{W}^{(c)}\bar{\mathbf{x}}_t + \mathbf{b}^{(c)}\big) \quad \text{for } t = 1, \dots, T,$$

where $\bar{\mathbf{x}}_t \in \mathbb{R}^F$, $\mathbf{W}^{(c)} \in \mathbb{R}^{N_c \times F}$, $\mathbf{b}^{(c)} \in \mathbb{R}^{N_c}$, and $\phi = \text{ReLU}$.

MaxPooling1D (pool size 1).

Pool size 1 leaves temporal resolution unchanged:

$$\mathbf{P}_t = \mathbf{C}_t.$$

Bidirectional LSTM (hidden units $H = 128$, return_sequences=True).

The BiLSTM processes $\{\mathbf{P}_t\}_{t=1}^T$ producing forward and backward hidden states:

$$\overrightarrow{\mathbf{h}}_t = \text{LSTM}_{\rightarrow}(\mathbf{P}_t, \overrightarrow{\mathbf{h}}_{t-1}), \qquad \overleftarrow{\mathbf{h}}_t = \text{LSTM}_{\leftarrow}(\mathbf{P}_t, \overleftarrow{\mathbf{h}}_{t+1}).$$

Concatenate to obtain the sequence of BiLSTM outputs:

$$\mathbf{H}_t = \begin{bmatrix} \overrightarrow{\mathbf{h}}_t \\ \overleftarrow{\mathbf{h}}_t \end{bmatrix} \in \mathbb{R}^{2H}, \quad t = 1, \dots, T.$$

Attention block (custom).

For each timestep we compute an unnormalized score and normalized attention weight, then the context vector:

$$\mathbf{u}_t = \tanh\big(\mathbf{W}^{(a)}\mathbf{H}_t + \mathbf{b}^{(a)}\big), \quad \mathbf{u}_t \in \mathbb{R}^{d_a}, \tag{20}$$

$$s_t = \mathbf{v}^\top \mathbf{u}_t + b^{(s)} \quad \in \mathbb{R}, \tag{21}$$

$$\alpha_t = \frac{\exp(s_t)}{\sum_{r=1}^T \exp(s_r)}, \qquad \sum_{t=1}^T \alpha_t = 1, \tag{22}$$

$$\mathbf{c} = \sum_{t=1}^T \alpha_t \mathbf{H}_t \in \mathbb{R}^{2H}. \tag{23}$$

Interpretation: α_t is the attention weight for timestep t, $\mathbf{c}$ is the attention-pooled representation.

Dense layers and dropout.

The context $\mathbf{c}$ passes through dense transformations with ReLU and dropout $p = 0.3$:

$$\mathbf{z}_1 = \phi\big(\mathbf{W}^{(1)}\mathbf{c} + \mathbf{b}^{(1)}\big), \tag{24}$$

$$\tilde{\mathbf{z}}_1 = \mathbf{d} \odot \mathbf{z}_1, \tag{25}$$

where $\mathbf{W}^{(1)} \in \mathbb{R}^{128\times 2H}$, $\phi = \text{ReLU}$, $\mathbf{d}$ is a Bernoulli mask with $\Pr(d_i = 1) = 1-p$ and $\odot$ denotes element-wise product. During inference an appropriate scaling is applied. The branch1 output:

$$\mathbf{b}_1 = \phi\big(\mathbf{W}^{(2)}\tilde{\mathbf{z}}_1 + \mathbf{b}^{(2)}\big) \in \mathbb{R}^{64}.$$

Branch 2: Direct Dense Path (Aggregated Feature Modeling)

This branch is the direct capture of global, non-temporal interactions between features. The flattened input goes through 128 units of dense layer with ReLU activation and dropout is used to avoid overfitting. A second dense layer of 64 units permits an increased degree of abstraction of feature relationships. This direction is a complement to the temporal one in the fact that it models the static aggregate behavior like volume, directionality, and intensity of traffic.

Flatten input across time and features:

$$\mathbf{f} = \text{Flatten}(\bar{\mathbf{X}}) \in \mathbb{R}^{F\cdot T}.$$

Two dense layers with dropout produce branch2 output:

$$\mathbf{g}_1 = \phi\big(\mathbf{W}^{(3)}\mathbf{f} + \mathbf{b}^{(3)}\big), \tag{26}$$

$$\tilde{\mathbf{g}}_1 = \mathbf{d}' \odot \mathbf{g}_1, \tag{27}$$

$$\mathbf{b}_2 = \phi\big(\mathbf{W}^{(4)}\tilde{\mathbf{g}}_1 + \mathbf{b}^{(4)}\big) \in \mathbb{R}^{64}. \tag{28}$$

3.4.3 Feature Fusion and Output Layer

The results of the two branches are combined to give a single high-level representation which combines both temporal sequences and aggregated behaviors. This combined information is again synthesized in a post-merge dense layer which has 64 units and dropout. The resulting classification is obtained after the 6 attack categories, namely, the Basic, DoS, Reconnaissance, Exploits, Fuzzers, and Backdoor are generated by a softmax layer. This is trained with the Adam optimizer, categorical crossentropy loss, class-weighting, and a batch size of 256 and 100 training epochs. The accuracy, precision, recall, and F1-score are assessment metrics to guarantee a multifaceted evaluation of the detection performance based on classes under imbalance.

Concatenate branch outputs:

$$\mathbf{m} = \begin{bmatrix}\mathbf{b}_1\\ \mathbf{b}_2\end{bmatrix} \in \mathbb{R}^{128}.$$

Post-merge dense + dropout:

$$\mathbf{h} = \phi\big(\mathbf{W}^{(5)}\mathbf{m} + \mathbf{b}^{(5)}\big) \in \mathbb{R}^{64}, \tag{29}$$

$$\tilde{\mathbf{h}} = \mathbf{d}'' \odot \mathbf{h}. \tag{30}$$

Final logits and softmax:

$$\mathbf{o} = \mathbf{W}^{(o)}\tilde{\mathbf{h}} + \mathbf{b}^{(o)} \in \mathbb{R}^{C}, \qquad \hat{\mathbf{y}} = \mathrm{softmax}(\mathbf{o}),$$

where C is the number of classes.

Loss with Class Weights

Using one-hot ground-truth $\mathbf{y} \in \{0,1\}^C$ and per-class weights w_c, the weighted categorical cross-entropy for a single example:

$$\mathcal{L}(\hat{\mathbf{y}}, \mathbf{y}) = -\sum_{c=1}^{C} w_c\, y_c \log(\hat{y}_c).$$

Batch loss is the mean over batch elements.

Optimization: Adam

Let θ denote the set of all trainable parameters. Adam maintains first and second moment estimates:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\nabla_\theta \mathcal{L}_t, \tag{31}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)(\nabla_\theta \mathcal{L}_t)^2, \tag{32}$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \quad \hat{v}_t = \frac{v_t}{1-\beta_2^t}, \tag{33}$$

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}. \tag{34}$$

Regularization and Metrics

Dropout acts as multiplicative Bernoulli noise during training, and weight decay (if used) adds $\lambda\|\theta\|_2^2$ to the loss. Performance is measured using Accuracy, Precision, Recall and F1 computed over the validation/test sets.

Remarks on the $T = 1$ case

When $T = 1$ the BiLSTM and attention reduce to per-sample transformations: the BiLSTM still maps the single-step input to a hidden vector and attention degenerates to a learned projection. The architectural design however preserves the ability to handle longer sequences if $T > 1$.

Table 5 Model Hyperparameters and Configuration Details

Parameter	**Value / Setting**	**Notes**
Feature Scaling	StandardScaler	Applied on X_train, X_val, X_test
Input Reshape	(samples, 1, features)	For Conv1D + LSTM
Target Encoding	One-hot (to_categorical)	Number of classes = len(le.classes_)
Conv1D filters	64	Kernel size = 1, Activation = ReLU
MaxPooling1D pool size	1	No dimensionality reduction
BiLSTM units	128	return_sequences=True
Attention	Dense + Multiply + Lambda sum	Custom attention block
Dense Layer 1 (Branch 1)	128 units, ReLU	After attention
Dropout (Branch 1)	0.3	After Dense layer
Dense Layer 2 (Branch 1)	64 units, ReLU	Branch output
Flatten	-	Directly flatten input
Dense Layer 1 (Branch 2)	128 units, ReLU	After flatten
Dropout (Branch 2)	0.3	After Dense layer
Dense Layer 2 (Branch 2)	64 units, ReLU	Branch output
Concatenate	-	Merge branch1 + branch2
Post-Merge Dense	64 units, ReLU	Post-merge
Post-Merge Dropout	0.3	Post-merge
Output Dense Layer	Units = #classes, softmax	Classification output
Optimizer	Adam	Default learning rate
Loss Function	Categorical Crossentropy	For multi-class classification
Evaluation Metrics	Accuracy, Precision, Recall, F1	Custom Keras metrics
Epochs	100	Total iterations
Batch Size	256	Number of samples per batch
Class Weight	{0:5, 1:1, 2:3, 3:6, 4:1, 5:5}	To handle class imbalance

The configuration details presented in Table 5 describe the entire process of preprocessing, architecture, and training values of the proposed hybrid intrusion detection model. The table indicates the standardization and re-shape of the features before they are loaded into the dual-branch network, and then the Conv1D, BiLSTM, Attention and Dense features are set in detail in the two branches. It also stores the vital training parameters like optimizer selection, loss function, metrics, classes weights, epochs and batch size. A combination of these parameters results in reproducibility and points to the ability of the model to acquire temporal, semantic, and aggregate feature patterns successfully.

4 Results and Discussion

The developed hybrid multi-branch architecture has been tested on the resulting fused CICIDS-2017 + UNSW-NB15 dataset, and the obtained results indicate good generalization capacity in different categories of heterogeneous attacks. The model also showed a high training accuracy of 96.80 percent, F1 of 96.83 percent, and a low training loss of 0.2408 during training and the validation accuracy and F1 both were 96.77 percent and 96.80 percent respectively. This small difference between the training and validation measures corresponds to high regularization and proves that the attention-based dual-branch feature extractor affects overfitting. This is in contrast to the typical CNN-only or LSTM-only IDS models which tend to have a severe drift when trained on multi-distribution fused datasets.

The model on the test dataset gave 89.88 percent accuracy, 0.9012 preciseness, 0.8970 recall, and 0.8990 F1-score, which establishes strong results on unseen traffic. Given the fact that the test set is mixed and contains minorities attack types, the macro-F1 of approximately 90 percent proves novelty and power of the proposed learning pipeline based on fusion. The results of the individual class-wise performance demonstrate a consistent high level of behavior: DoS (0.93 F1), Backdoor (0.91 F1), and Reconnaissance (0.90 F1) were identified with a high degree of reliability, which confirms the ability of the model to recognize temporal-statistical features even of visual-similar attack flows. A marginally less good performance of the Normal class (0.88 F1) is caused by overlapping traffic patterns with benign-appearing reconnaissance probes which is often a difficulty in real network IDS research.

The stability of classes was also increased with threshold-tuning. The results of utilizing adaptive threshold optimization to the validation set showed that the macro-F1 was 0.90, whereas Normal, Fuzzers, and Exploits classes showed to have a better balance of false positives and false negatives. This illustrates the significance of probability calibration in the anomaly detection process of multi-distribution features in which the boundaries of classes can change when fused datasets are used.

Table 6 overviews the training and validation outcomes of proposed model. The accuracy and F1-score are high and almost equal in both stages, which indicates a high degree of generalization and the low overfitting. The fact that the validation loss is low, once again confirms the strength of the hybrid architecture over single-branch deep IDS models.

Table 6 Training and Validation Performance of the Proposed Hybrid Model

Metric	Accuracy	Precision	Recall	F1-score
Training	0.9680	0.9706	0.9659	0.9683
Validation	0.9677	0.9700	0.9660	0.9680

Training and validation results demonstrate strong convergence and balanced learning.

Table 7 presents the test-set evaluation. The hybrid model attains almost 90 per cent by all measures, which proves its capacity to generalize between unseen traffic of both CICIDS-2017 and UNSW-NB15. The precision-recall balance is used to make sure that the model is not biased towards the attack and benign classes.

Table 8 emphasizes individual performance by class. The majority of the types of attacks have F1-scores of greater than 0.88, with DoS, Backdoor, and Reconnaissance representing the highest detection. There is a slightly low recall of the Normal class because they overlap with low-volume reconnaissance traffic. These findings indicate how effectively the hybrid feature extractor can single-handedly detect subtle patterns of attacks, even with very heterogeneous fused traffic.

Table 7 Test-Set Evaluation Metrics

Metric	Value	Interpretation	Remark
Accuracy	0.8988	Overall correctness	High reliability
Precision	0.9012	FP control	Stable predictions
Recall	0.8970	FN control	Robust detection
F1-score	0.8990	Harmonic mean	Balanced performance

Table 8 Class-wise Performance of the Proposed Model

Class	Precision	Recall	F1-score
Backdoor	0.90	0.92	0.91
DoS	0.93	0.93	0.93
Exploits	0.90	0.87	0.89
Fuzzers	0.85	0.91	0.88
Normal	0.93	0.84	0.88
Reconnaissance	0.88	0.92	0.90

The process of combining the CICIDS-2017 and UNSW-NB15 datasets with the help of the suggested data-fusion strategy allowed the intrusion detection framework to improve its generalization power considerably. Commonly, traditional IDS models which were trained on single-source data may be distributional biased, and may not exhibit the variety of attacks; but the heterogeneous fusion of network traffic proved to offer a more realistic and rich source of both modern and legacy attack behaviors. The fused data by dissolving schema inconsistencies, matching statistical flow representations and removing redundant or noisy features allowed the hybrid architecture to acquire consistent patterns of invariance across a wide variety of settings. The stable validation accuracy (96.77%) and strong test performance (90% macro-F1) of this model showed that the architecture was able to scale to multi-distribution traffic.

This was further enhanced by the dual-branch CNN-BiLSTM Attention architecture which harnesses complementary feature space on the basis of fused data. The convolutional branch learned local flow dependency existing in both datasets, whereas the recurrent branch learned temporal properties of progressive attack patterns. The attention mechanism was able to promote discriminability by emphasizing feature interactions that were cross-dataset features that most strongly predicts the presence of malicious behavior. Such a collaboration between augmented fused data and optimized architecture gave powerful class-wise performance, particularly in difficult attack categories, like Backdoor and Reconnaissance. All in all, the presented fusion-based architecture depicts a new and efficient method of holistic intrusion detection on the heterogeneous network domains.

5 Conclusion

This paper presented a new hybrid multi-branch architecture of intrusion detection that has the ability to generalize to heterogeneous network conditions. The proposed framework has solved the problem of distributional bias in single-dataset-trained IDS models by combining CICIDS-2017 and UNSW-NB15 datasets into a single feature space. The fusion approach increased the malicious flows with statistical, temporal, and behavioral variety, which allowed the model to acquire resilient and transferable attack patterns. The attention-based dual-branch CNN BiLSTM extractor was able to effectively learn local and long-range temporal dependencies to extract signal of local flow, achieving high-level discriminability than other single-branch deep learning networks. The effectiveness of the approach is proven through empirical results. The close similarity in the F1-scores of 96.80 and 96.77 percent and training and validation accuracies of 96.80 and 96.77 percent indicate strong convergence and low levels of overfitting. The test set performance of using 89.88 percentage accuracy and macro-F1 of 0.8990 confirm that the architecture is reliable to predict the unseen fused traffic. Class-based analysis can also be used to underscore the strength of the model especially in identifying difficult types of attacks like DoS, Backdoor, and Reconnaissance. The adaptability of the system in multi-distribution settings was supported by the fact that threshold calibration enhanced the precision recall balance to complex classes. Despite the high performance of the proposed system, there are still some potential directions available to improve the system. To start with, introduction of federated learning can facilitate decentralized implementation of IDS in distributed networks and maintain privacy of data. Second, the fusion pipeline should be extended to incorporate real-time traffic, IoT, or cloud telemetry data so that they can be applied in more multi-layered infrastructures. Third, studying explainability approaches, like SHAP or attention heatmaps, could be another way of gaining a deeper understanding of the attribution of the attack, which would aid SOC analysts decision-making. Also, it can be further improved by the use of transformer-based encoders or CodeBERT-like architectures to enhance cross-domain generalization and temporal modeling. Lastly, an adaptive online-learning module might enable the IDS to keep up with changes in the threats and decrease concept drift due to dynamic environments.

References

1. I. Rakine et al., Comprehensive Review of Intrusion Detection Techniques: ML and DL in Different Networks, IEEE Access, vol. 13, pp. 104345 104367, 2025, doi: 10.1109/ACCESS.2025.3579990.
2. D. Patil, Artificial Intelligence In Cybersecurity: Enhancing Threat Detection And Prevention Mechanisms Through Machine Learning And Data Analytics, Dec. 2024, doi: 10.2139/SSRN.5057410.
3. M. Mohy-Eddine, A. Guezzaz, S. Benkirane, M. Azrour, and Y. Farhaoui, An Ensemble Learning Based Intrusion Detection Model for Industrial IoT Security, Big Data Min. Anal., vol. 6, no. 3, pp. 273 287, Sep. 2023, doi: 10.26599/BDMA.2022.9020032.

4. D. Jain et al., ASA-LSTM-based brain tumor segmentation and classification in MRI images, Int. J. Adv. Technol. Eng. Explor., vol. 11, no. 115, pp. 838 851, Jun. 2024, doi: 10.19101/IJATEE.2023.10102143.
5. R. Gopi et al., Intelligent Intrusion Detection System for Industrial Internet of Things Environment , doi: 10.32604/csse.2023.025216.
6. A. Hussain, E. Marin Tordera, X. Masip-Bruin, and H. C. Leligou, Rule-Based With Machine Learning IDS for DDoS Attack Detection in Cyber-Physical Production Systems (CPPS), IEEE Access, vol. 12, pp. 114894 114911, 2024, doi: 10.1109/ACCESS.2024.3445261.
7. G. B. Gaggero, A. Armellin, G. Portomauro, and M. Marchese, Industrial Control System-Anomaly Detection Dataset (ICS-ADD) for Cyber-Physical Security Monitoring in Smart Industry Environments, IEEE Access, vol. 12, pp. 64140 64149, 2024, doi: 10.1109/ACCESS.2024.3395991.
8. M. Hatti, IoT-Enabled Energy Efficiency Assessment of Renewable Energy Systems and Micro-Grids in Smart Cities : Harnessing the Power of IoT to Create Sustainable and Efficient Urban Environments, Volume 2, p. 522, 2025.
9. J. S. Kushwah, D. Jain, P. Singh, A. K. Pandey, S. Das, and P. Vats, A Comprehensive System for Detecting Profound Tiredness for Automobile Drivers Using a CNN, Lect. Notes Electr. Eng., vol. 914, pp. 407 415, 2022, doi: 10.1007/978-981-19-2980-9_33.
10. A. Thakkar and R. Lohiya, A Review on Challenges and Future Research Directions for Machine Learning-Based Intrusion Detection System., Arch. Comput. Methods Eng., vol. 30, no. 7, p. 4245, Sep. 2023, doi: 10.1007/S11831-023-09943-8.
11. E. Alonso et al., On the Detection Capabilities of Signature-Based Intrusion Detection Systems in the Context of Web Attacks, Appl. Sci. 2022, Vol. 12, Page 852, vol. 12, no. 2, p. 852, Jan. 2022, doi: 10.3390/APP12020852.
12. J. S. Kushwah, D. Gupta, A. Shrivastava, P. Ambily Pramitha, J. T. Abraham, and M. Lunagaria, Analysis and visualization of proxy caching using LRU, AVL tree and BST with supervised machine learning, Mater. Today Proc., vol. 51, pp. 750 755, Jan. 2022, doi: 10.1016/J.MATPR.2021.06.224
13. Sen et al., On specification-based cyber attack detection in smart grids, Energy Informatics 2022 51, vol. 5, no. 1, pp. 23-, Sep. 2022, doi: 10.1186/S42162-022-00206-7.
14. D. Ribeiro dos Santos, Specification-based Intrusion Detection for Hierarchical Hybrid Industrial Control Systems, p. 255, Apr. 2024, doi: 10.34894/VQ1DJA.
15. A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, Machine Learning and Deep Learning Approaches for CyberSecurity: A Review, IEEE Access, vol. 10, pp. 19572 19585, 2022, doi: 10.1109/ACCESS.2022.3151248.
16. D. Musleh, M. Alotaibi, F. Alhaidari, A. Rahman, and R. M. Mohammad, Intrusion Detection System Using Feature Extraction with Machine Learning Algorithms in IoT, J. Sens. Actuator Networks 2023, Vol. 12, Page 29, vol. 12, no. 2, p. 29, Mar. 2023, doi: 10.3390/JSAN12020029.
17. G. Logeswari, S. Bose, and T. Anitha, An Intrusion Detection System for SDN Using Machine Learning , doi: 10.32604/iasc.2023.026769.

18. S. M. Kasongo, A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework, Comput. Commun., vol. 199, pp. 113 125, Feb. 2023, doi: 10.1016/J.COMCOM.2022.12.010.
19. E. Altulaihan, M. A. Almaiah, and A. Aljughaiman, Anomaly Detection IDS for Detecting DoS Attacks in IoT Networks Based on Machine Learning Algorithms, Sensors 2024, Vol. 24, Page 713, vol. 24, no. 2, p. 713, Jan. 2024, doi: 10.3390/S24020713.
20. S. Soliman, W. Oudah, and A. Aljuhani, Deep learning-based intrusion detection approach for securing industrial Internet of Things, Alexandria Eng. J., vol. 81, pp. 371 383, Oct. 2023, doi: 10.1016/J.AEJ.2023.09.023.
21. T. E. Ali, Y. W. Chong, and S. Manickam, Comparison of ML/DL Approaches for Detecting DDoS Attacks in SDN, Appl. Sci. 2023, Vol. 13, Page 3033, vol. 13, no. 5, p. 3033, Feb. 2023, doi: 10.3390/APP13053033.
22. R. A. Abed, E. K. Hamza, and A. J. Humaidi, A modified CNN-IDS model for enhancing the efficacy of intrusion detection system, Meas. Sensors, vol. 35, p. 101299, Oct. 2024, doi: 10.1016/J.MEASEN.2024.101299.
23. A. Awajan, A Novel Deep Learning-Based Intrusion Detection System for IoT Networks, Comput. 2023, Vol. 12, Page 34, vol. 12, no. 2, p. 34, Feb. 2023, doi: 10.3390/COMPUTERS12020034.
24. I. Ullah and Q. H. Mahmoud, Design and Development of RNN Anomaly Detection Model for IoT Networks, IEEE Access, vol. 10, pp. 62722 62750, 2022, doi: 10.1109/ACCESS.2022.3176317.
25. B. Deore and S. Bhosale, Intrusion Detection System Based on RNN Classifier for Feature Reduction, SN Comput. Sci. 2021, vol. 3, no. 2, pp. 114-, Dec. 2021, doi: 10.1007/S42979-021-00991-0.
26. Singh, Manali, Jitendra Singh Kushwah, Yogendra Rathore, and Kirti Shrivastava. ”The Study of Swarm Intelligence Technique: A Review.” Grenze International Journal of Engineering & Technology (GIJET) 10 (2024).
27. A. A. E. B. Donkol, A. G. Hafez, A. I. Hussein, and M. M. Mabrook, Optimization of Intrusion Detection Using Likely Point PSO and Enhanced LSTM-RNN Hybrid Technique in Communication Networks, IEEE Access, vol. 11, pp. 9469 9482, 2023, doi: 10.1109/ACCESS.2023.3240109.
28. T. Kim and W. Pak, Early Detection of Network Intrusions Using a GAN-Based One-Class Classifier, IEEE Access, vol. 10, pp. 119357 119367, 2022, doi: 10.1109/ACCESS.2022.3221400.
29. S. Rahman, S. Pal, S. Mittal, T. Chawla, and C. Karmakar, SYN-GAN: A robust intrusion detection system using GAN-based synthetic data for IoT security, Internet of Things, vol. 26, p. 101212, Jul. 2024, doi: 10.1016/J.IOT.2024.101212.
30. F. Yan, G. Zhang, D. Zhang, X. Sun, B. Hou, and N. Yu, TL-CNN-IDS: transfer learning-based intrusion detection system using convolutional neural network, J. Supercomput., vol. 79, no. 15, pp. 17562 17584, Oct. 2023, doi: 10.1007/S11227-023-05347-4.
31. O. D. Okey, D. C. Melgarejo, M. Saadi, R. L. Rosa, J. H. Kleinschmidt, and D. Z. Rodriguez, Transfer Learning Approach to IDS on Cloud IoT Devices Using Optimized CNN, IEEE Access, vol. 11, pp. 1023 1038, 2023, doi: 10.1109/ACCESS.2022.3233775.

32. S. Racherla, P. Sripathi, N. Faruqui, M. Alamgir Kabir, M. Whaiduzzaman, and S. Aziz Shah, Deep-IDS: A Real-Time Intrusion Detector for IoT Nodes Using Deep Learning, IEEE Access, vol. 12, no. May, pp. 63584 63597, 2024, doi: 10.1109/ACCESS.2024.3396461.
33. A. Talukder, A. Layek, and A. Hossain, ACU-Net : Attention-based convolutional U-Net model for segmenting brain tumors in fMRI images, 2025, doi: 10.1177/20552076251320288.

    .

34. J. Singh Kushwah, A. Kumar, S. Patel, R. Soni, A. Gawande, and S. Gupta, Comparative study of regressor and classifier with decision tree using modern tools, Mater. Today Proc., vol. 56, pp. 3571 3576, Jan. 2022, doi: 10.1016/J.MATPR.2021.11.635.
35. M. M. Aslam, A. Tufail, L. C. De Silva, and R. A. A. H. M. Apong, Multi-Feature Hybrid Anomaly Detection in ICS: An Integration of ML, DL, and Statistical Techniques, ACM SecTL 2025 - Proc. 3rd ACM Work. Secur. Trust. Deep Learn. Syst. Part ACM AsiaCCS 2025, pp. 43 51, Aug. 2025, doi: 10.1145/3709021.3737669.
36. F. R. Alzaabi and A. Mehmood, A Review of Recent Advances, Challenges, and Opportunities in Malicious Insider Threat Detection Using Machine Learning Methods, IEEE Access, vol. 12, pp. 30907 30927, 2024, doi: 10.1109/ACCESS.2024.3369906.
37. A. S, S. D, and P. G, Malicious insider threat detection using variation of sampling methods for anomaly detection in cloud environment, Comput. Electr. Eng., vol. 105, p. 108519, Jan. 2023, doi: 10.1016/J.COMPELECENG.2022.108519.
38. N. Jeffrey, Q. Tan, and J. R. Villar, A Review of Anomaly Detection Strategies to Detect Threats to Cyber-Physical Systems, Electron. 2023, Vol. 12, Page 3283, vol. 12, no. 15, p. 3283, Jul. 2023, doi: 10.3390/ELECTRONICS12153283.
39. S. Rawat, A. Srinivasan, V. Ravi, and U. Ghosh, Intrusion detection systems using classical machine learning techniques vs integrated unsupervised feature learning and deep neural network, Internet Technol. Lett., vol. 5, no. 1, p. e232, Jan. 2022, doi: 10.1002/ITL2.232.
40. E. Jaw, X. Wang, R. Alcantud, and L. Jontschi, Feature Selection and Ensemble-Based Intrusion Detection System: An Efficient and Comprehensive Approach, Symmetry 2021, Vol. 13, Page 1764, vol. 13, no. 10, p. 1764, Sep. 2021, doi: 10.3390/SYM13101764.
41. M. Chora and M. Pawlicki, Intrusion detection approach based on optimised artificial neural network, Neurocomputing, vol. 452, pp. 705 715, Sep. 2021, doi: 10.1016/J.NEUCOM.2020.07.138.
42. Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, Benchmarking of Machine Learning for Anomaly Based Intrusion Detection Systems in the CICIDS2017 Dataset, IEEE Access, vol. 9, pp. 22351 22370, 2021, doi: 10.1109/ACCESS.2021.3056614.
43. S. Hajj, R. El Sibai, J. Bou Abdo, J. Demerjian, A. Makhoul, and C. Guyeux, Anomaly-based intrusion detection systems: The requirements, methods, measurements, and datasets, Trans. Emerg. Telecommun. Technol., vol. 32, no. 4, p. e4240, Apr. 2021, doi: 10.1002/ETT.4240.

44. A. H. Azizan et al., A Machine Learning Approach for Improving the Performance of Network Intrusion Detection Systems, Ann. Emerg. Technol. Comput., vol. 5, no. 5, pp. 201 208, Mar. 2021, doi: 10.33166/AETiC.2021.05.025.
45. L. Ashiku and C. Dagli, Network Intrusion Detection System using Deep Learning, Procedia Comput. Sci., vol. 185, pp. 239 247, Jan. 2021, doi: 10.1016/J.PROCS.2021.05.025.
46. P. Vanin et al., A Study of Network Intrusion Detection Systems Using Artificial Intelligence/Machine Learning, Appl. Sci. 2022, Vol. 12, Page 11752, vol. 12, no. 22, p. 11752, Nov. 2022, doi: 10.3390/APP122211752.
47. E. Alhajjar, P. Maxwell, and N. Bastian, Adversarial machine learning in Network Intrusion Detection Systems, Expert Syst. Appl., vol. 186, p. 115782, Dec. 2021, doi: 10.1016/J.ESWA.2021.115782.
48. M. Sajid et al., Enhancing intrusion detection: a hybrid machine and deep learning approach, J. Cloud Comput. 2024 131, vol. 13, no. 1, pp. 123-, Jul. 2024, doi: 10.1186/S13677-024-00685-X.
49. A. Aldallal, Toward Efficient Intrusion Detection System Using Hybrid Deep Learning Approach, Symmetry 2022, Vol. 14, Page 1916, vol. 14, no. 9, p. 1916, Sep. 2022, doi: 10.3390/SYM14091916.
50. E. U. H. Qazi, M. H. Faheem, and T. Zia, HDLNIDS: Hybrid Deep-Learning-Based Network Intrusion Detection System, Appl. Sci. 2023, Vol. 13, Page 4921, vol. 13, no. 8, p. 4921, Apr. 2023, doi: 10.3390/APP13084921.
51. M. A. Onsu, M. Simsek, M. Fobert, and B. Kantarci, Intelligent multi-sensor fusion and anomaly detection in vehicles via deep learning, Internet of Things, vol. 31, p. 101561, May 2025, doi: 10.1016/J.IOT.2025.101561.
52. H. Zhang, D. Upadhyay, M. Zaman, A. Jain, and S. Sampalli, SC-MLIDS: Fusion-based Machine Learning Framework for Intrusion Detection in Wireless Sensor Networks, Ad Hoc Networks, vol. 175, p. 103871, Aug. 2025, doi: 10.1016/J.ADHOC.2025.103871.
53. C. Zhang, P. Hu, and L. Tan, Spatiotemporal Feature Correlation with Feature Space Transformation for Intrusion Detection, Appl. Sci. 2025, Vol. 15, Page 11168, vol. 15, no. 20, p. 11168, Oct. 2025, doi: 10.3390/APP152011168.
54. P. Tyagi, D. A. K. Bindal, and D. Srivastava, An Optimized Feature-Level Fusion Framework for Multimodal Biometric Authentication Using ML Classifiers, Int. J. Environ. Sci., vol. 11, no. 7s, pp. 120 135, Jun. 2025, doi: 10.64252/JMZ3X190.
55. M. Chora Micha and Pawlicki, Intrusion detection approach based on optimised artificial neural network, Neurocomputing, vol. 452, pp. 705 715, Sep. 2021, doi: 10.1016/J.NEUCOM.2020.07.138.
56. D. Palma and P. L. Montessoro, A Post-Quantum Cryptography Enabled Feature-Level Fusion Framework for Privacy-Preserving Multimodal Biometric Recognition, Cryptogr. 2025, Vol. 9, Page 72, vol. 9, no. 4, p. 72, Nov. 2025, doi: 10.3390/CRYPTOGRAPHY9040072.
57. O. P. Ayeoribe, O. Akinsanmi, and I. V. Ayeoribe, Multi-Sensor Fusion for Improved Accuracy in Wireless Intrusion Detection, SSRN Electron. J., Jun. 2025, doi: 10.2139/SSRN.5368184.