Iraqi Journal of Intelligent Computing and Informatics (IJICI)
Vol. 4, 2. December 2025, pp. 227~238
ISSN: 2791-2868, DOI: 10.52940/ijici.v4i2.134

AI-BASED INTRUSION DETECTION FRAMEWORK

ALI RAED AL-SULTANI [1]

[1] Department of Electrical and Computer Engineering, Altinbaş University. Turkey

Article Info

Article history:

Received October 18, 2025
Revised December 21, 2025
Accepted December 30, 2025

Keywords:

Autoencoder,
BiLSTM with Attention,
CNN-LSTM,
Cyber Threat Detection,
Hybrid Models,
Intrusion Detection System (IDS).

ABSTRACT

The increasing pace of cyber threats has enormous threats to the confidentiality, integrity, and availability of a network. Defense Intrusion Detection System (IDS) is one of the most important defense mechanisms and traditional signature-based systems have problems with unheard-before attacks. The article presents an artificial-intelligence (AI)-based IDS that integrates three hybrid deep learning models, which are a Convolutional Neural NetworkLong Short-Term Memory (CNN-LSTM) network, a Bidirectional LSTM with attention mechanism (BiLSTM Attention) network, and an Autoencoder with a Random Forest (AERF) classifier. The models are tested on NSL-KDD and CICIDS2017 benchmark datasets after a single preprocessing pipeline, which incorporates data cleaning, normalization, categorical features encoding, statistical feature selection and Synthetic Minority Over-Sampling Technique (SMOTE)-based balancing of classes. On both datasets, experimental results demonstrate almost perfect recall and F1-scores. CNN-LSTM, AERF and a combination of the three models have high and complimentary performance in various categories of attacks. The high detection rates and the low false-alarm rates are validated through Receiver Operating Characteristic (ROC) and Precision Recall (PR) curves along with confusion matrices. Overall, the proposed framework demonstrates strong scalability and generalization capability, highlighting its potential for deployment in modern network cybersecurity environments.

Corresponding Author:

ALI RAED AL-SULTANI
Department of Electrical and Computer Engineering, Altinbaş University. Turkey
Email: ali.raed.eng@gmail.com

1. INTRODUCTION

Internet has become an essential part of everyday life however its expansion has also led to the exposure of vulnerability to malware attacks, eavesdropping and sophisticated cyberattacks [1]-[3]. Cyber intrusions endanger the confidentiality, integrity and availability (CIA) of networked systems and may start with reconnaissance and vulnerability scanning followed by the exploitation of vulnerability found [4][5]. Current attacks are password brute force attacks (methodical trial and error guessing of credentials), botnets, DDoS attacks, and cross-site scripting (XSS) attacks, and are reported to take place on a huge scope, with malware being identified around 13 seconds and cybercrime expenses estimated to reach 9.5 trillion dollars by 2024 [7][8].

Intrusion Detection Systems (IDSs) have hence been a critical part of network defense. Classical misuse-based IDSs are based on known signatures and only useful to attack well characterized but vulnerable to new or changing threats [9]. Instead, anomaly-based IDSs model normal behavior and signal deviations and can identify intrusions never seen before at the price of increased false-alarms. Anomaly-based IDSs are more and more using statistical, knowledge-based or machine learning (ML) and deep learning (DL) technologies [10], with the former possessing the capability to detect features based on supervised, unsupervised, or semi-

supervised approaches [11] and the latter having more scalability and the capability to learn complex feature representations presented by data [12][13].

Benchmark datasets like KDD99, NSL-KDD, UNSW-NB15 as well as CICIDS2017 [14][15] are important to assess IDS models by availing standardized traffic traces and various attack situations. Of these, CICIDS2017 is especially appreciated in the context of its extensive and realistic coverage of attacks, with NSL-KDD being the most popular, being a sophisticated, refined version of KDD99 that alleviates many of its known shortcomings. However, most current IDS methods have been designed to either single dataset or a small number of attacks, where they usually have shallow modelizations or simplified preprocessing pipelines, that fail to consider the noisy features, class imbalance and high dimensional features space. To overcome these threats, this paper suggests an AI-based IDS system that relies on hybrid deep learning systems. The framework integrates a highly-detailed preprocessing chain with three compatible hybrid solutions and an integration approach involving en ensembles, the objective of enhancing detection and minimizing false positives in a variety of and changing threats on both NSL-KDD and CICIDS2017.

The principal findings of this paper are as follows:

- Complete preprocessing pipeline: We come up with a unified preprocessing pipeline consisting of data cleaning, normalization, categorical features encoding, statistical features selection based on Mutual Information and Chi-Square tests, and SMOTE-based class balancing, to enhance the performance of model learning on both NSL-KDD and CICIDS2017.
- Three hybrid deep learning frameworks of IDS: We create and test three hybrid IDS, CNN-LSTM, BiLSTM With Attention and (AERF)t that collaborate to utilize spatial, temporal and latent features representations to improve intrusion detection.
- Ensemble integration strategy: We propose an ensemble integration strategy (A + B + C) that integrates the performance of the three proposed models to use their complementary capabilities and provide more balanced and reliable performance in detection.
- Vast empirical analysis along two metrics: We experiment heavily on both NSL-KDD and CICIDS2017, and report accuracy, precision, recall, F1-score, ROC and PR curves, as well as confusion matrices, and we demonstrate that the proposed hybrid and ensemble methods can provide very high classification accuracy.
- Comparison to baseline ML classifiers: To confirm the efficiency of the proposed framework, we compare its performance to the baseline (ML) classifiers and find that in all the evaluation metrics, there are improvements.

The rest of this paper is structured in the following way. Section II evaluates the literature concerning (ML) and (DL) in intrusion detection. Section III describes the methodology proposed such as the preparation of the datasets, preprocessing, feature selection, and the design of hybrid models. Section IV provides the experimental environment and results on both NSL-KDD and CICIDS2017, whereas Section V provides training curves, performance comparisons, and visual analyses. Section VI is the conclusion of the paper, that summarizes the findings of the paper and future work directions.

2. RELATED WORK

Researchers have proposed diverse ML/DL-based IDSs across IoT, big data, vehicular, and WSN environments. For IoT, [16] introduced a hybrid feature selection method with fuzzy TOPSIS and DGWA, while [17] designed a CNN-WDLSTM hybrid for big data. HyDL-IDS for CAN networks achieved near-perfect accuracy in [18], and HIIDS combining Spark MLlib with LSTM-autoencoder was presented in [19]. A multi-phase IoT IDS integrating LSTM, CNN, (AE), and SMOTE was proposed in [20], while [21] developed HIDS-RPL for IoMT with 99.87% accuracy. (AE) based botnet detection was shown in [22], DBN for CICIDS2017 in [23], and stacked ResNet models in [24]. Other works include IoT-IDCS-CNN leveraging GPUs [25], ML-STL with Tomek balancing [26], FA-ML with Support Vector Machine (SVM) and Grey Wolf Optimizer [27], LOA-CatBoost for WSN [28], CNN+RNN for DoS [29], and a DL framework with EABOA [30]. Additional approaches explored IoT modeling with SMOTE and cyclic encoding [31], semi-supervised IDS usfAD for unseen attacks [32], ensemble models for multiclass IoT IDS [33], feature extraction/selection trade-offs [34], federated IDS with privacy methods [35], and a two-tier OCC IDS for zero-day threats [36].

The above summaries of studied studies prove that the use of ML and DL methods can be effectively used in detecting intrusion in different settings, such as the IoT, in-vehicle, medical, and industrial systems [16]-[36]. Many of them are specific to a type of network or dataset, they commonly optimize a single model on a specific situation (e.g., CAN bus traffic, RPL routing, or an individual IoT dataset). They claim high accuracy, but one

can notice several limitations. To begin with, much of the previous research uses only one benchmark dataset or small sets of attack taxonomies, and this calls into question cross-dataset generalization. Second, feature engineering and preprocessing pipelines are not specified in detail, and the choice of features, the effect of feature imbalance, and the effect of various preprocessing steps on the detection accuracy are rarely analyzed. Third, in spite of the fact that hybrid and ensemble models are already studied, comparatively few systematic comparisons have been conducted on multiple hybrid DL architectures and their integration into a single IDS system.

Conversely, the current research aims at the classical NSL-KDD, and the contemporary CICIDS2017 datasets, which is pre-processed in a more in-depth and unified pipeline and creates three complementing hybrid models, which are assessed independently and in pool. The design can help us to compare the behaviour of various architectural options (CNN-LSTM, BiLSTM-Attention and AE-RF) when subjected to identical experimental conditions, and to measure the improvement of ensemble integration over hybrid models as well as traditional ML baselines. In turn, the proposed work is expected to address the issue of the lack of correspondence between dataset-specific solutions and more general AI-based IDS frameworks.

3. PROPOSED METHODOLOGY

The suggested AI-driven IDS architecture is based on a multi-stage pipeline of raw records of network traffic to final intrusion labels. Initial, raw data of NSL-KDD and CICIDS2017 are processed by means of data cleaning, missing value management, numerical features normalization, and categorical features encoding. Second, there is feature selection step using Mutual Information and Chi-Square ranking that selects the most informative features and hopefully dimensions are also reduced. Third, the issue of class imbalance is reduced by using SMOTE to sample the training sets to create artificial samples to minority attack classes. Fourth, the selected and balanced amounts of feature vectors are presented to three hybrid detection models, which are CNN-LSTM, BiLSTM-Attention, and AE-RF. Both models provide probabilities of classes under normal and attack. Lastly, to make the final decision, these outputs are summed up by an ensemble (overall average results or majority rule). Such an end-to-end process correlates input traffic facts to predicted labels, which offers a broad IDS paradigm across other datasets.

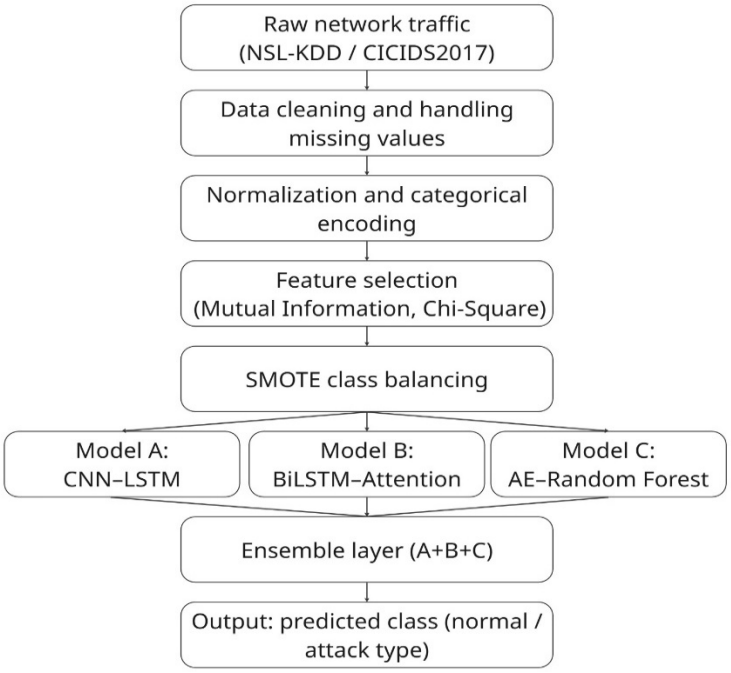

Figure 1: Overall architecture of the proposed AI-based IDS framework, illustrating the preprocessing pipeline, hybrid models, and ensemble decision stage.

The entire process of the proposed AI-based IDS is shown in Figure 1. This step starts with uncoded network traffic of the NSL-KDD and CICIDS2017 datasets that are processed sequentially through several preprocessing steps, such as data cleaning, missing values, normalization, and categorical encoding. The statistical selection (Mutual Information and Chi-Square) is then applied to extract relevant features, then class balancing using SMOTE is applied to overcome the problem of imbalance in the dataset. The processed data is then inputted into three parallel hybrid detection models including CNN-LSTM, BiLSTM with Attention

and (AERF). Their results are combined using ensemble layer that sums the strength of all the three models. As shown in the figure, this structured pipeline allows a strong and generalizable intrusion detection framework that can be used to reach high accuracy and consistent classification of various attacks.

3.1. Methodological Formulation

The proposed IDS framework can be formalized as a supervised learning problem in which Xdenotes the set of network flow instances represented by selected features, and $Y$the corresponding set of class labels. The objective is to learn a decision function:

$$f: X \rightarrow Y$$

That maximizes detection performance across all attack categories, including low-frequency intrusions typically underrepresented in real-world traffic. For each instance x, the three hybrid models produce probability distributions $p_A(x), p_B(x), p_C(x)$, which are then integrated through an ensemble mechanism. The final prediction is computed either through averaging,

$$p_{\text{ens}}(x) = \frac{1}{3}\,(p_A(x) + p_B(x) + p_C(x)),$$

The hybrid-ensemble architecture will stabilize the prediction results of the three models by averaging the prediction interests of the three models by probability or majority voting, thus formulating a robust end-to-end learning goal. The hybrid models were chosen based on their complementary inductive biases: CNNs are local spatial dependencies, LSTM and BiLSTM units are sequential and bidirectional temporal dependencies, attention mechanisms highlight salient features, (AEs) are non-linear dimensionality reductions, and (RF) are ensemble robustness. The combination of these various views gives the ensemble strength over individual models, and generates more successful and coherent detection results in different traffic conditions. Moreover, the preprocessing pipeline, such as feature selection and SMOTE-based balancing, is used to reduce the dimensions and enhance the quality of the representations, whereas the computational efficiency of CNN-LSTM, BiLSTM-Attention, and (AERF) is used to guarantee scalability. Together, this combined design facilitates high generalization and successful identification of common and uncommon intrusions.

3.2. NSL-KDD DATASET

The NSL-KDD dataset, an enhanced version of the KDD Cup 1999 benchmark, comprises 41 traffic features describing individual network connections. These features include numerical attributes (e.g., connection duration, byte counts, and various traffic statistics) as well as nominal attributes (e.g., protocol type, service, and TCP flag). Each record is labeled as either normal or one of four attack categories: Denial of Service (DoS), Probe, Remote-to-Local (R2L), or User-to-Root (U2R).

1. Data preprocessing

The initial training and test partitions were initially combined to reach one large dataset. The categorical variables (protocol, service and flag) were numerically represented and the labels of classes were mapped on their respective attack type. The feature-wise mean values were used to impute the numerical features that had missing values and infinity. All numeric variables were then scaled to counterbalance the differences in the scale to ensure that large scale variables like the number of bytes were not overpowering the learning.

2. Exploratory data analysis (EDA)

The exploratory analysis showed that there were high class imbalance: the normal traffic and DoS attacks were most common and U2R and R2L were extremely low. The TCP was also dominant in the traffic and User Datagram Protocol (UDP) and Internet Control Message Protocol (ICMP) was much less in frequency. Such an imbalance pointed out to a high possibility of bias to majority classes and was the reason to resort to the application of re-sampling methods to get a more balanced training distribution.

3. Balancing with SMOTE

In order to reduce the impact of classes imbalance, SMOTE was used on the training data. SMOTE creates new synthetic minority samples based on interpolation of existing samples in the feature space. This step enhanced the proportion of the under-represented attacks and created a more balanced distribution of classes, thus allowing the models to learn discriminative patterns of the frequent and rare attacks and enlarging the effective training set significantly.

4. Feature selection

The purpose of the feature selection was to enhance the computational efficiency and less redundancy, maintaining discriminative information in this way. Two statistical ranking criteria, complementary to each other were applied:

- **Mutual Information (MI):** MI is a measure of the dependence between each feature and the class label, and it measures both linear and non-linear dependence. The features with a high MI score can give more information about the target variable and thus they are said to be more relevant.
- **Chi-Square test:** Chi-Square statistic is used to determine the extent of correlation between categorical features and the class label. Characteristics that have high Chi-Square are highly dependent on the class and could be included well.

The SMOTE-balanced data was subjected to both types of ranking and features that were important in one of the criteria were kept. As the first target range of 20 to 30 features was taken into account, the joint selection was 39 features. This tradeoff ensures statistical significance and information diversity, but silences noisy or redundant variables, therefore making training more efficient and potentially enhancing generalization performance.

5. Hybrid deep learning models

Three hybrid deep learning architectures were evaluated on NSL-KDD to exploit complementary modeling capabilities and capture both structural and sequential properties of network flows.

- **Model A CNNLSTM hybrid:** this model is a combination of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) units. The CNN layers are used to obtain local patterns and correlations between features whereas LSTM layers are used to model temporal relationships across sequence of connections. Mechanisms of regularization like dropout and batch normalization are also included to make it robust. The design of this model allows it to model local feature interactions as well as long-range temporal patterns of traffic data.
- **Model B - BiLSTM with attention:** The second model is a BiLSTM with attention. BiLSTM sequentially processes both forward and backward sequences, and it records the past and future dependencies. The attention layer gives the time steps or features that are salient dynamic weights to the important parts of the sequence. This enhances sensitivity of the model to another subtle attack signature besides interpretability of its decisions.
- **Model C -Autoencoder and Random Forest:** This third architecture will combine a neural autoencoder with a (RF) classifier. The (AE) reduces the dimensionality of the high-dimensional feature space to a small latent representation, which is non-linear dimensionality reduction and noisy samples. These latent vectors are then submitted to the RF classifier which yields final labels. This combines (DL) as the representation learner and ensemble techniques as a robust decision-maker, which is useful in noisy or initially unbalanced environments.

When combined, the three models complement each other since CNN-LSTM detects both spatiotemporal patterns, the BiLSTM-Attention model emphasizes the sequential relationship and identifies salient features, whereas the (AERF) hybrid applies both unsupervised representations learning and the process of classification using an ensemble. Their relative comparison of NSL-KDD indicates that hybrid architectures may improve the rate of detection of frequent and rare attacks.

3.3 CICIDS2017 DATASET

CICIDS2017 dataset offers a realistic framework of intrusion detection and simulates real traffic where benign flows and various attacks such as DDoS, brute force, botnets, infiltration, and web threats are represented. It is very representative of the current intrusion patterns and behaviors because of its rich numerical and categorical characteristics.

1. Data Preprocessing

The preparation of the data entailed the encoding of categorical variables (e.g., protocol fields) and target labels followed by the division of the data into training, validation, and test data sets by stratified sampling to maintain the class distribution. Any missing or invalid numerical values were substituted with feature averages and scales were normalized, such as the flow duration, and the number of packets to ensure that no feature dominated the learning process.

2. Exploratory Data Analysis (EDA)

Analysis of the data revealed excessive imbalance in the classes with normal traffic and common attacks prevailing over the rare intrusions which were few, which could lead to biasness towards common classes. The wide range of attributes of the dataset both basic flow data and detailed statistics provide high discriminatory power, but need some preprocessing to maintain both common and rare patterns to model well.

3. Balancing with SMOTE

Synthetic oversampling was implemented to balance the training set to create new samples of the minority classes by interpolation. This created in excess of 80,000 samples per category, which corrects the bias in favor of typical traffic, and also allows models to be more capable of detecting frequent and uncommon attacks.

4. Feature Selection

In the case of CICIDS2017, feature filtering has been used to trim noise, decrease dimension and enhance the performance of models. Since redundant and weakly related attributes were numerous, selection was done with two complementary statistics ranking methods:

- **Mutual Information (MI):** It is a technique that identifies both linear and non-linear relationships between features and the target in which the high values of the MI indicate stronger relevance to differentiating attacks and normal traffic.
- **Chi-Square Test:** This is a statistical test that measures the relationship between features and the target whereby the higher the chi-square the larger the dependence of classes and predictive quality.

Features that were ranked highest in both methods were integrated and 46 attributes were selected as opposed to the original 20-30 attributes. The complexity was minimized without affecting the discriminative power in this preprocessing stage and this increased model learning and generalization in order to improve intrusion detection which is more robust.

3.4 Hybrid Deep Learning Models

Three hybrid deep learning models were evaluated on CICIDS2017 to determine the effectiveness with regard to modern traffic. They have integrated spatial, temporal, and latent feature representations to provide strong intrusion detection of all types of attacks.

- **Model A CNN-LSTM Hybrid:** The former model is a hybrid of CNNs and LSTMs: CNNs identify local patterns in features, whereas the LSTMs identify sequential dynamics. This structural and temporal learning integration makes it handy when it comes to the detection of attacks that evolve with time.
- **Model B BiLSTM with Attention:** The second model is the combination of BiLSTM and attention, and it handles the sequences in both directions to compute the past and future dependencies. The attention layer emphasizes important features and time steps and enhances detection accuracy and interpretability.
- **Model C – Autoencoder and Random Forest:** The third model combines an autoencoder and a (RF). The (AE) consists of reducing noise and preserving the important information in the high-dimensional CICIDS2017 features by compressing them into latent representations and classifying the latent representation with the help of the (RF). This is a hybrid combining (DL) feature extraction with ensemble robustness, which provides good performance in dealing with complex, imbalanced data.

The three models present complementary insights into the intrusion detection in CICIDS2017: CNN-LSTM identifies spatiotemporal patterns, BiLSTM focuses on attention in each sequence and highlights key features, whereas the (AERF) hybrid integrates deep representation learning with ensemble classification. They serve in combination to present a complete end-to-end architecture of the analysis of the large-scale intrusion detection data of today.

Algorithm 1 Hybrid Deep Learning Models for Intrusion Detection

Preprocessed feature matrix X, class labels Y

Trained models M_A, M_B, M_C

function TRAIN_CNN_LSTM(X, Y)

$X_{seq} \leftarrow \text{reshape}(X)$

$F_{cnn} \leftarrow \text{CNN\_Layers}(X_{seq})$

$F_{lstm} \leftarrow \text{LSTM\_Layers}(F_{cnn})$

$y_{\text{pred}} \leftarrow \text{Dense}_{softmax}(F_{lstm})$

Train network using cross-entropy loss

return M_A

end function

function TRAIN_BILSTM_ATTENTION(X, Y)

$X_{seq} \leftarrow \text{reshape}(X)$

$F_{bi} \leftarrow \text{BiLSTM\_Layers}(X_{seq})$

$\alpha \leftarrow \text{Attention}(F_{bi})$

$C \leftarrow \sum_t \alpha_t \cdot F_{bi,t}$

$y_{\text{pred}} \leftarrow \text{Dense}_{softmax}(C)$

Train network using cross-entropy loss

return M_B

end function

function TRAIN_AE_RF(X, Y)

$Z \leftarrow \text{Encoder}(X)$

$\hat{X} \leftarrow \text{Decoder}(Z)$

Train autoencoder using MSE loss

Train Random Forest classifier on Z

return M_C

end function

$M_A \leftarrow \text{TRAIN\_CNN\_LSTM}(X, Y)$

$M_B \leftarrow \text{TRAIN\_BILSTM\_ATTENTION}(X, Y)$

$M_C \leftarrow \text{TRAIN\_AE\_RF}(X, Y)$

return M_A, M_B, M_C

4. RESULTS AND DISCUSSION

The three models CNN-LSTM, BiLSTM with Attention, and (AERF) were trained and evaluated on both NSL-KDD and CICIDS2017 to assess their effectiveness. All experiments were conducted under the same conditions for fair comparison, with performance measured using accuracy, precision, recall, F1-score, and ROC-AUC. Confusion matrices were also analyzed to evaluate strengths and weaknesses in detecting minority attack classes. The detailed results are presented in the following tables and figures for both datasets.

4.1. NSL-KDD RESULTS

Table 1 presents the performance comparison of CNN-LSTM, BiLSTM with Attention, and (AERF) on the NSL-KDD dataset, evaluated using accuracy, precision, recall, F1-score, and ROC-AUC.

Table 1. Performance comparison of hybrid models on NSL-KDD Dataset

Model	Accuracy	Precision	Recall	F1	ROC-AUC
CNN-LSTM	0.990473	0.786991	0.826638	0.794007	N/A
AE + RF	0.995522	0.745135	0.730712	0.729085	N/A
BiLSTM + Attention	0.989631	0.728466	0.731342	0.714498	N/A

(AERF) model has the highest overall accuracy (99.55%), and a lower F1-score (0.729) than CNN–LSTM, which indicates less performance on minority classes but a high one on majority classes. CNN-LSTM offers

the most accurate balance of metrics with the highest F1-score (0.794), and recall (0.827) which is essential in the case of intrusion detection where false alarms are expensive. BiLSTM with Attention has similar accuracy (98.96) but lower F1 (0.714) indicating that, in the case of NSL-KDD, the convolutional-recurrent architecture is more successful with capturing discriminative patterns compared to the exclusively recurrent architecture with attention. On the whole, each of the three hybrids is effective; however, CNN-LSTM has the best trade-off between the detection rate and the overall classification performance.

Table 1 extends the analysis to include the ensemble (A+B+C) alongside the individual hybrid models.

Table 2. Performance comparison of hybrid and ensemble models on NSL-KDD Dataset

Model	Accuracy	Precision	Recall	F1	ROC-AUC
Ensemble (A+B+C)	0.993974	0.835687	0.812084	0.804646	N/A
CNN-LSTM	0.990473	0.786991	0.826638	0.794007	N/A
AE + RF	0.995522	0.745135	0.730712	0.729085	N/A
BiLSTM + Attention	0.989631	0.728466	0.731342	0.714498	N/A

The ensemble model has the best precision (0.836) and F1-score (0.805), which implies that the three hybrid models are better than any single model. Although the (AERF) has the best accuracy, it has the lowest F1-score, indicating that the accuracy is not a sufficient parameter to define the performance of an IDS when there is imbalance in the classes. CNN-LSTM is the best single model as it has better recall, and the BiLSTM with Attention is lower than the CNN-LSTM in most of the metrics. These findings indicate that the hybrid models can be better assembled to enhance the robustness as well as lessening the tradeoff between accuracy and recall on NSL-KDD.

4.2 CICIDS2017 RESULTS

Table 3 presents the results of the hybrid models, baseline classifiers, and their combinations on the CICIDS2017 dataset, evaluated using accuracy, precision, recall, F1-score, and ROC-AUC as performance metrics.

Table 3. Performance comparison of hybrid and baseline models on CICIDS2017 Dataset

Model	Accuracy	Precision	Recall	F1	ROC-AUC
AE + RF	0.999911	0.999922	0.999922	0.999922	1.000000
RF(base)	0.999911	0.999961	0.999883	0.999922	1.000000
CNN-LSTM	0.999491	0.999766	0.999336	0.999551	0.999998
MLP (baseline)	0.999402	0.999766	0.999180	0.999473	0.999999
BiLSTM + Attention	0.999203	0.999648	0.998946	0.999297	0.999972
SVM (baseline)	0.998361	0.998244	0.998867	0.998555	0.999397

All the models in CICIDS2017 have extremely high values in all metrics, and F1-scores along with ROC-AUCs are close to 1.0. The (AERF) and the control (RF) have nearly the same performance with both having a F1-score of 0.9999 and ROC-AUC of 1.0, which emphasizes the power of tree-based ensembles on this highly-featured data. The CNN-LSTM and the MLP baseline also show good results with F1-scores of more than 0.9994, whereas the BiLSTM with Attention and SVM have slightly lower results but still in a very small margin of performance. The differences between models on CICIDS2017 are smaller than between NSL-KDD, indicating that the feature space is more abundant and the sample size is larger, which is subsequently associated with an easier classification environment when using proper preprocessing.

Table 4. Performance comparison of hybrid, ensemble, and baseline models on CICIDS2017

Model	Accuracy	Precision	Recall	F1	ROC-AUC
AE + RF	0.9999	0.9999	0.9999	0.9999	1.0000
RF(base)	0.9999	0.99996	0.99988	0.9999	1.0000
CNN-LSTM	0.9995	0.99977	0.99934	0.99955	0.99999
Ensemble (A+B+C)	0.9995	0.99944	0.99952	0.99948	N/A
MLP (baseline)	0.9994	0.99977	0.99918	0.99947	0.99999
BiLSTM + Attention	0.9992	0.99965	0.99895	0.99930	0.99997
SVM (baseline)	0.9984	0.99824	0.99887	0.99856	0.99940

The (AERF) and the baseline (RF) have the best scores, which means that this set of features and a tree-based classifier is especially effective in CICIDS2017. CNN-LSTM and MLP baseline are immediately followed, and BiLSTM-Attention and SVM are slightly behind them. The ensemble (A+B+C) achieves a high performance that is similar to the best single models, but performance improvement is relatively small when compared to isolated hybrids as most models are already running with very high performance. This is in contrast to NSL-KDD implying that ensembling is most useful in more difficult settings where there is a strong class imbalance and less separable features.

4.3 Discussion

The hybrid models across both datasets possess complementary strengths, with the relative performance of the models being dependent on the complexity of the dataset and class imbalances. CNN-LSTM offers the most suitable balance of recall and precision on NSLKDD, whereas the (AERF) is the highest classification but at the cost of minority classes; the convolutional feature extraction advantage is somewhat shown by the BiLSTM with Attention. The ensemble model provides the most consistent overall performance that supports the advantage of mixing heterogeneous systems in problematic environments. On CICIDS2017, the gap between the performance of the classifiers is significantly reduced, with the greatest performance of the tree-based classifiers, especially the (AERF), but the deep learning models also excel in their work. Although these are the strengths, the assessment is restricted on the offline supervised experiments and does not consider dynamic conditions of traffic, concept drift, and adversarial behavior. In the future, the work undertaking will be centered towards real-time deployment and robustness analysis and adaptive learning strategies to make the work more applicable in operational settings.

5. EXPERIMENTAL SETUP

The NSL-KDD and CICIDS2017 datasets were experimented using the preprocessing pipeline outlined in Section III that comprises of data cleaning, data normalization and categorical encoding, feature selection using Mutual Information and Chi-Square tests, and SMOTE-based class balancing. To maintain the class distributions, the three hybrid models (CNN- LSTM, BiLSTM with Attention, and (AERF) and the baseline classifiers (RF, SVM and MLP) were trained with stratified splits. The maximum epoching of training was done up to 30 epochs with early stopping and dropout regularization to prevent overfitting. The performance of models was measured based on accuracy, precision, recall, F1-score, and ROC-AUC, and the experiments were conducted in the same conditions to be able to compare them fairly.

5.1 NSL-KDD DATASET

- **Metric Comparison Bar Charts**

The accuracy of AE + RF, Ensemble (A + B + C), CNN -LSTM, BiLSTM + Attentions on NSL-KDD are shown in Figure 2. All the models are highly accurate with a minor difference thus point to a high predictive ability in all the approaches. The CNN-LSTM and ensemble in particular are doing fairly well as they are indicative of the advantages of product-complementary model combination and convolutional-recurrent extraction of features.

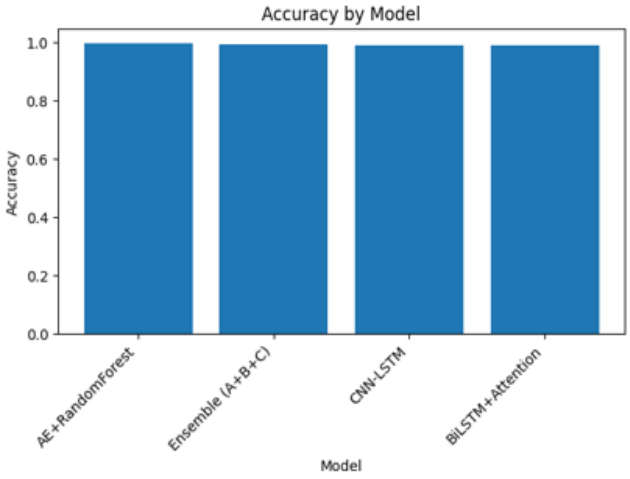

Figure 2: Accuracy by Model.

5.2 CICIDS2017 DATASET

- **Metric and Curve Analysis**

In CICIDS2017 more graphical analysis is also given. The binary ROC curve of CNN-LSTM, the BiLSTM + Attention, the AE + RF, and the ensemble (A+B+C) are presented in Figure 3. The curves are all near the upper-left corner and all the values of AUC are very close to 1 and this means that the benign and malicious traffic can be perfectly separated as well as the false-positive rates are very low over a large range of thresholds.

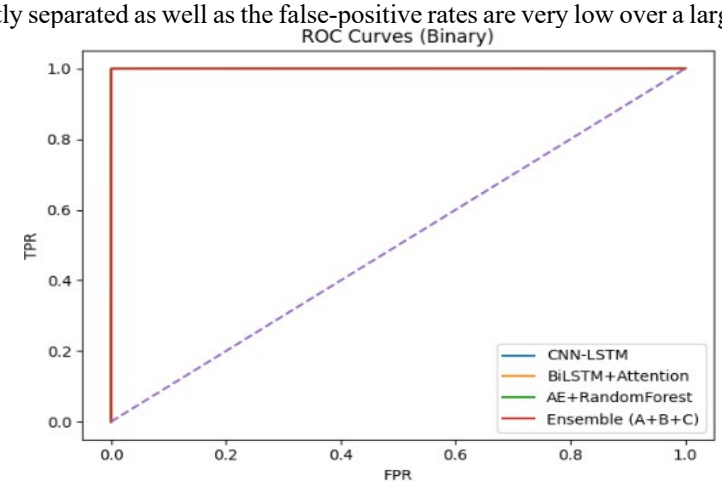

Figure 3: ROC Curves.

- **Precision-Recall Curves (Binary)**

The figure 4 shows the precision-recall curves of the same models. Both the precision and the recall are close to 1 and the values of average precision are close to or equal to 1.0. The curves are near the upper line. These findings prove that the models are highly accurate but do not compromise high recall, hence reducing false alarm and miss detections in such a binary context.

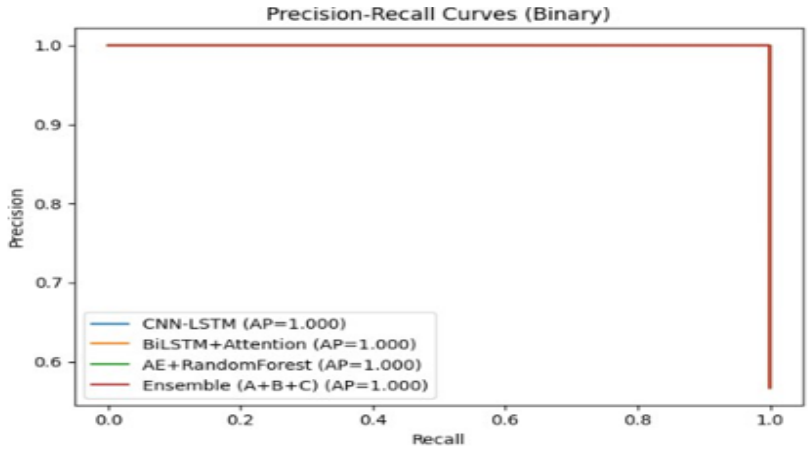

Figure 4: Precision-Recall Curves (Binary).

Though the suggested models perform very well on both NSL-KDD and CICIDS2017, the models are only tested in offline supervised learning with controlled data divisions. The real network conditions are subject to changing traffic patterns, concept drift, and limited computational resources that might have an effect on the performance. Future research will thus aim to implement lightweight versions of the models in real-time applications and research on their resilience to adversarial examples as well as stealthy attacks.

6. CONCLUSION

The current paper presented an intrusion detection framework based on the three hybrid deep learning models, namely CNN-LSTM, BiLSTM with Attention, and (AERF), and tested it on the NSLKDD and CICIDS2017 benchmark datasets. The empirical findings indicate that although each of the individual models has a good performance, the combination of the models has the best-balanced performance in NSL-KDD as there is class imbalance and randomly distributed attack patterns thus increasing the difficulty of the task performed. CNN-LSTM has the optimal recall F1 trade autosomely and the accuracy of (AERF) is high in all cases, and the ensemble can take advantage of their complementary strengths to enhance their robustness. Both hybrid and baseline models have an extremely high performance on CICIDS2017, with the tree-based methods slightly outperforming the rest. Generally, the paper validates the authenticity of hybrid and ensemble methods in the modern intrusion detection. Future research will focus on generalising the framework to real-time IoT/IIoT systems, adapting and resource-sensitive model versions, and exploring robustness to adversarial and changing attack conditions.

REFERENCES

[1] TechTarget, ‘‘6 common types of cyber attacks and how to prevent them,’’ 2023. Accessed: 22 August 2025.
[2] BlackBerry, ‘‘Quarterly global threat report—september 2024,’’ 2024. Accessed: 22 August 2025.
[3] U. Tariq, I. Ahmed, A. K. Bashir, and K. Shaukat, ‘‘A critical cybersecurity analysis and future research directions for the internet of things: A compre- hensive review,’’ Sensors, vol. 23, no. 8, p. 4117, 2023.
[4] M. Conti, T. Dargahi, and A. Dehghantanha, ‘‘Cyber threat intelligence: challenges and opportunities,’’ Cyber threat intelligence, pp. 1–6, 2018.
[5] O. Faker and E. Dogdu, ‘‘Intrusion detection using big data and deep learn- ing techniques,’’ in Proceedings of the 2019 ACM Southeast conference,pp. 86–93, 2019.
[6] G. Kaur, A. H. Lashkari, and A. Rahali, ‘‘Intrusion traffic detec- tion and characterization using deep image learning,’’ in 2020 IEEE Intl Conf on dependable, autonomic and secure computing, Intl Conf on pervasive intelligence and computing, Intl Conf on cloud and big data computing, Intl Conf on cyber science and technology congress (DASC/PiCom/CBDCom/CyberSciTech), pp. 55–62, IEEE, 2020.
[7] Internet Security Threat Report, ‘‘Internet security threat report,’’ 2018. Accessed: 22 August 2025.
[8] Cyersecurity Ventures, ‘‘Cybercrime to cost the world $9 trillion annually in 2024,’’ 2024. Accessed: 22 August 2025.
[9] P. Wu, Deep learning for network intrusion detection: Attack recognition with computational intelligence. PhD thesis, UNSW Sydney, 2020.
[10] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, ‘‘Survey of intrusion detection systems: techniques, datasets and challenges,’’ Cyber- security, vol. 2, no. 1, pp. 1–22, 2019.
[11] N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, ‘‘Survey on sdn based network intrusion detection system using machine learning ap- proaches,’’ Peer-to-Peer Networking and Applications, vol. 12, no. 2,pp. 493–501, 2019.

[12] R. Vinayakumar, M. Alazab, K. P. Soman, P. Poornachandran, A. Al- Nemrat, and S. Venkatraman, ‘‘Deep learning approach for intelligent intrusion detection system,’’ IEEE access, vol. 7, pp. 41525–41550, 2019.
[13] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, ‘‘A survey of deep learning-based network anomaly detection,’’ Cluster Computing, vol. 22, no. Suppl 1, pp. 949–961, 2019.
[14] G. C. Fernández and S. Xu, ‘‘A case study on using deep learning for network intrusion detection,’’ in MILCOM 2019-2019 IEEE Military Com- munications Conference (MILCOM), pp. 1–6, IEEE, 2019.
[15] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, ‘‘A survey of network-based intrusion detection data sets,’’ Computers & security, vol. 86, pp. 147–167, 2019.
[16] Y. Pourardebil Khah, M. Hosseini Shirvani, and H. Motameni, ‘‘A hybrid machine learning approach for feature selection in designing intrusion detection systems (ids) model for distributed computing networks,’’ The Journal of Supercomputing, vol. 81, no. 1, p. 254, 2025.
[17] M. M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian, and G. Fortino, ‘‘A hybrid deep learning model for efficient intrusion detection in big data environment,’’ Information Sciences, vol. 513, pp. 386–396, 2020.
[18] W. Lo, H. Alqahtani, K. Thakur, A. Almadhor, S. Chander, and G. Kumar, ‘‘A hybrid deep learning based intrusion detection system using spatial- temporal representation of in-vehicle network traffic,’’ Vehicular Commu- nications, vol. 35, p. 100471, 2022.
[19] M. A. Khan and Y. Kim, ‘‘Deep learning-based hybrid intelligent intrusion detection system.,’’ Computers, Materials & Continua, vol. 68, no. 1, 2021.
[20] B. Susilo, A. Muis, and R. F. Sari, ‘‘Intelligent intrusion detection system against various attacks based on a hybrid deep learning algorithm,’’ Sen- sors, vol. 25, no. 2, p. 580, 2025.
[21] A. Berguiga, A. Harchay, and A. Massaoudi, ‘‘Hids-rpl: A hybrid deep learning-based intrusion detection system for rpl in internet of medical thing networks,’’ IEEE Access, 2025.
[22] Y. N. Soe, Y. Feng, P. I. Santosa, R. Hartanto, and K. Sakurai, ‘‘Machine learning-based iot-botnet attack detection with sequential architecture,’’ Sensors, vol. 20, no. 16, p. 4372, 2020.
[23] S. Manimurugan, S. Al-Mutairi, M. M. Aborokbah, N. Chilamkurti,
S. Ganesan, and R. Patan, ‘‘Effective attack detection in internet of medical things smart environment using a deep belief neural network,’’ Ieee Access, vol. 8, pp. 77396–77404, 2020.
[24] B. Alotaibi and M. Alotaibi, ‘‘A stacked deep learning approach for iot cyberattack detection,’’ Journal of Sensors, vol. 2020, no. 1, p. 8828591, 2020.
[25] Q. Abu Al-Haija and S. Zein-Sabatto, ‘‘An efficient deep-learning-based detection and classification system for cyber-attacks in iot communication networks,’’ Electronics, vol. 9, no. 12, p. 2152, 2020.
[26] M. A. Talukder, S. Sharmin, M. A. Uddin, M. M. Islam, and S. Aryal, ‘‘Mlstl-wsn: machine learning-based intrusion detection using smotetomek in wsns,’’ International Journal of Information Security, vol. 23, no. 3,pp. 2139–2158, 2024.
[27] M. Karthikeyan, D. Manimegalai, and K. RajaGopal, ‘‘Firefly algorithm based wsn-iot security enhancement with machine learning for intrusion detection,’’ Scientific Reports, vol. 14, no. 1, p. 231, 2024.
[28] S. S. Abinayaa, P. Arumugam, D. B. Mohan, A. Rajendran, A. Lashab, B. Wei, and J. M. Guerrero, ‘‘Securing the edge: Catboost classifier op- timized by the lyrebird algorithm to detect denial of service attacks in internet of things-based wireless sensor networks,’’ Future Internet, vol. 16, no. 10, p. 381, 2024.
[29] S. Salmi and L. Oughdir, ‘‘Performance evaluation of deep learning tech- niques for dos attacks detection in wireless sensor network,’’ Journal of Big Data, vol. 10, no. 1, p. 17, 2023.
[30] S. Dontu, R. Vallabhaneni, S. R. Addula, P. K. Pareek, and R. R. Hussein, ‘‘Enhanced adaptive butterfly optimizer based feature selection for pro- tecting the data in industry based wsn,’’ in 2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS), pp. 1–6, IEEE, 2024.
[31] Z. Cao, Z. Zhao, W. Shang, S. Ai, and S. Shen, ‘‘Using the ton-iot dataset to develop a new intrusion detection system for industrial iot devices,’’ Multimedia Tools and Applications, pp. 1–29, 2024.
[32] M. A. Uddin, S. Aryal, M. R. Bouadjenek, M. Al-Hawawreh, and M. A. Talukder, ‘‘usfad based effective unknown attack detection focused ids framework,’’ Scientific Reports, vol. 14, no. 1, p. 29103, 2024.
[33] S. Hajla, E. Ennaji, Y. Maleh, and S. Mounir, ‘‘Enhancing iot network defense: advanced intrusion detection via ensemble learning techniques,’’ Indones. J. Electr. Eng. Comput. Sci, vol. 35, no. 3, pp. 2010–2020, 2024.
[34] J. Li, M. S. Othman, H. Chen, and L. M. Yusuf, ‘‘Optimizing iot intrusion detection system: feature selection versus feature extraction in machine learning,’’ Journal of Big Data, vol. 11, no. 1, p. 36, 2024.
[35] D. Torre, A. Chennamaneni, J. Jo, G. Vyas, and B. Sabrsula, ‘‘Toward en- hancing privacy preservation of a federated learning cnn intrusion detection system in iot: method and empirical study,’’ ACM Transactions on Software Engineering and Methodology, vol. 34, no. 2, pp. 1–48, 2025.
[36] M. A. Uddin, S. Aryal, M. R. Bouadjenek, M. Al-Hawawreh, and M. A. Talukder, ‘‘A dual-tier adaptive one-class classification ids for emerging cyberthreats,’’ Computer Communications, vol. 229, p. 108006, 2025.