Journal of Theoretical and Applied Information Technology
15th January 2025. Vol.103. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

HYBRID DEEP LEARNING FRAMEWORK FOR INTRUSION DETECTION: INTEGRATING CNN, LSTM, AND ATTENTION MECHANISMS TO ENHANCE CYBERSECURITY

[1]A. HARSHAVARDHAN [2]DR. M. SREE VANI [3]DR ANITHA PATIL [4]NAGENDAR YAMSANI [5]KANDE ARCHANA

[1]Sr. Asst. Prof, Dept. of CSE (AIML&IoT), VNR Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana
[2]professor, Department Of CSE, Bvrit Hyderabad College Of Engineering For Women, Hyderabad -500090
[3] Professor Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Aziz Nagar Hyderabad-500075, Telangana, India.
[4]Assistant Professor, School of Computer Science and Artificial Intelligence, SR UNIVERSITY, Warangal, Telangana, India
[5]Assistant Professor, Department of Computer Science and Engineering, Sreenidhi University, Hyderabad.
[1]harshavgse@gmail.com [2]sreevani.m@bvrithyderabad.edu.In
[3]panitha243@gmail.com [4]nagendar.yamsani@gmail.com [5]kande.archana@gmail.com

ABSTRACT

Advanced intrusion detection systems (IDS) are required to protect contemporary networks due to the increasing complexity of cyber attacks. Due to their inability to fully capture the complex temporal and spatial patterns in network traffic, traditional intrusion detection systems (IDS) methods—such as standalone machine learning and deep learning models—frequently result in high false-positive rates and decreased detection accuracy. These drawbacks show how creative frameworks that can handle these problems are required. A hybrid deep learning framework that combines Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks and an attention mechanism is proposed in this study. The framework effectively captures spatial features, sequential dependencies, and critical network traffic patterns, enhancing accuracy and interpretability. The methodology includes comprehensive data preprocessing, principal component analysis (PCA) for dimensionality reduction, and recursive feature elimination (RFE) for feature selection. Hybrid Deep Learning-based Intrusion Detection (HDLID), a revolutionary algorithm, directs the suggested system's implementation. Tested on the UNSW-NB15 dataset, the framework outperforms state-of-the-art precision, recall, and F1-score models, achieving an impressive accuracy of 97.89%. The results validate its robustness and scalability for real-world applications. The proposed framework offers a practical, high-performance solution for intrusion detection, addressing limitations in existing methodologies and contributing to improved cybersecurity in diverse network environments.

Keywords - *Intrusion Detection System (IDS), Deep Learning, Cybersecurity, UNSW-NB15 Dataset, Network Threat Detection*

1. INTRODUCTION

With the expansion of interconnected systems and a growing surge of complexities within cyber threats, Dependability and adaptability are critical requirements for intrusion detection systems (IDS). Legacy cybersecurity strategies have a relatively poor ability to identify highly complicated and evolving attack patterns, contributing to high under-exploitation rates in network ecosystems. Big data, especially in healthcare, is also a significant concern. Deep learning has become the new solution to this problem by detecting any potential threats from the data. While considerable progress has been made, existing models often are hampered by deficiencies, including poor representation of spatial and temporal patterns, high false-positive rates, and problems with computational efficiency.

The research paper gives an overview of CNN, LSTM, and Hybrid models for feature extraction in IDS. 2.3 Studies highlight the effectiveness of deep learning-based models in feature learning and attack detection, but they also show limitations, including interpretability, scalability, and adaptability to real systems. In addition, recent reviews highlight the promise of hybrid deep learning architectures for tackling these issues, particularly architectures with attention capability. However, the full potential of such models remains unexplored, particularly in several vital capabilities — specifically, the ability to jointly combine spatial, sequential, and feature prioritization capabilities.

This work suggested a mixed deep-learning architecture for intrusion detection to close these gaps, including a CNN, an LSTM, and an attention mechanism. The objective is to construct a system that effectively captures spatial and temporal patterns while dynamically focusing on relevant features. The proposed research enhances recall, accuracy, precision, and F1-score techniques, outperforming the most advanced models. The main novelty is the w/o making the components interpret, which can generalize to both whys and different networks.

Although recent advancements in machine learning and deep-learning-based algorithms have improved their efficiency, the complexity and universality of intrusion detection and classification problems still result in several challenges in this domain. Earlier research employing isolated models, such as CNNs and LSTMs, excelled in spatial and temporal feature extraction, respectively, but struggled with interpretability and robustness against changing attack patterns. And yet, these methods typically suffer from high false-positive rates, restricting their practicality to real-world scenarios. To address these gaps, this research presents a new hybrid deep learning framework that integrates the strengths of CNN, LSTM, and attention for improved accuracy and interpretability. This paper scales well to modern times problems as it extracts spatial features using CNNs, temporal features using LSTMs, and gives priority to features using attention mechanism, thus achieving state-of-the-art performance metrics (accuracy: 97.89%, compared with the best models) for the current model and the past few models each of which have a limited input window.

This paper makes several contributions: it develops a novel hybrid model, evaluates its performance, and compares it to baseline models on the UNSW-NB15 dataset, showing better outcomes. The paper's structure is set up like this: Reviewing relevant literature, Section 2 finds possibilities and gaps. The suggested methodology, including feature engineering, data preparation, and model design, is explained in Section 3. Results from the experiment and a comparison are shown in Section 4. The results, their ramifications, and the study's limitations are covered in Section 5. Section 6 provides challenges and research issues. The study is finally concluded in Section 7, which also offers recommendations for further research focusing on scalability and relevance to actual intrusion detection.

2. RELATED WORK

The literature on deep learning-based cybersecurity highlights the significant advancements and diverse applications of intelligent frameworks while emphasizing the persistent challenges and emerging opportunities in designing robust intrusion detection systems for modern network environments. Combining big data and deep learning technologies has significantly enhanced IoT security by enabling scalable, intelligent systems to address cyber threats [1]. As discussed in [2], Federated learning approaches provide privacy-preserving solutions for cyber-attack detection in distributed environments. Several systematic reviews have highlighted the effectiveness of deep learning for IoT security [3], while comprehensive surveys [4] provide insights into databases and methodologies for systems for detecting intrusions (IDS). The robustness of ensemble deep learning techniques has also been investigated for detecting cyberattacks in industrial control systems [5].

Blockchain-based deep learning approaches, as in [6], and IoT analytics for innovative city development [7] highlight the diverse applications of deep learning in cybersecurity. The industrial IoT benefits from deep learning for monitoring and predictive maintenance [8], while reviews like [9] and [10] emphasize the need for innovative frameworks in network intrusion detection. Studies on network traffic monitoring and adversarial threats in IoT systems [11], [12] further underscore the evolving challenges in the domain. Technological reviews [13] have documented the advances in deep learning

algorithms, including hybrid models proposed for botnet detection in IoT networks [14]. IoT-based machine learning strategies [15], [16] continue to push boundaries, leveraging real-time analytics for proactive security. Recent taxonomies of deep learning techniques [17], applications in smart cities [18], and reinforcement learning approaches [19] further demonstrate the wide-ranging impact of AI-driven cybersecurity solutions.

Deep learning has been pivotal in 5G security [20] and predictive maintenance [21], with frameworks like distributed deep learning proving effective in IoT networks [22]. Novel techniques like zero-bias deep learning [23] and HealthFog systems for smart healthcare [24] have added new dimensions to intrusion detection. Federated deep learning mechanisms [25] and optimized algorithms for IoT security [26] are setting new standards for secure systems. Deep learning models for intrusion detection, including sequential frameworks [37], hybrid models [32], and class imbalance solutions [36], address critical limitations in traditional approaches. Novel architectures, such as LSTM-AE [32] and CNN-LSTM [33], highlight the growing sophistication of IDS frameworks. Comprehensive reviews [34], surveys [35], and advancements in SDN [38] demonstrate the field's progression.

Research on adversarial attack defense [40], optimization algorithms [39], and resilient IDS frameworks [37] has practical implications. Foundational studies on UNSW-NB15 [41], CNNs [42], LSTMs [43], and autoencoders [44] provide a basis for innovative models, while generative adversarial networks [45] and ensemble autoencoders [46] continue to drive progress in IDS methodologies. Collectively, these studies underscore the significance of hybrid deep learning models for improved cybersecurity.

Despite a lot of research work on deep learning for the intrusion detection, the limitation is still prevalent. In example, although standalone CNN-based IDS models have been proven to be precise in extracting spatial features, they cannot capture temporal relationships due to the lack of an inherent temporal modeling such as the use of temporal convolution and pooling networks (XYZ, (2023)). In contrast, IDS models based on LSTM have a good performance in terms of temporal sequence learning but are usually weak for high-dimensional network traffic data [16]. Hybrid methods (ABC, 2022) tried to combine CNN and LSTM, reach an accuracy of 94.75% but do not apply an attention mechanism, essential, in our opinion, to learn which features are more important in network traffic. Our study addresses these gaps by leveraging CNNs, LSTMs, and attention together in a unified framework, resulting in improved detection metrics while also providing improved interpretability and scalability as shown in Table 1.

3. PROPOSED FRAMEWORK

This paper's inherent feature, the deep learning-based IDS, uses machine learning to strengthen cybersecurity [3]. The process consists of data preprocessing, feature engineering, and a hybrid deep learning model that uses an attention mechanism in conjunction with CNNs and LSTM, as shown in Figure 1. The methodology covers data preparation using the UNSW-NB15 data set, feature optimization, model creation, and rigorous performance assessment. This architecture proposes a scaled, interpretable, and deployable system and successfully shows its efficiency in predicting diverse cyber-attacks and preventing them accordingly.

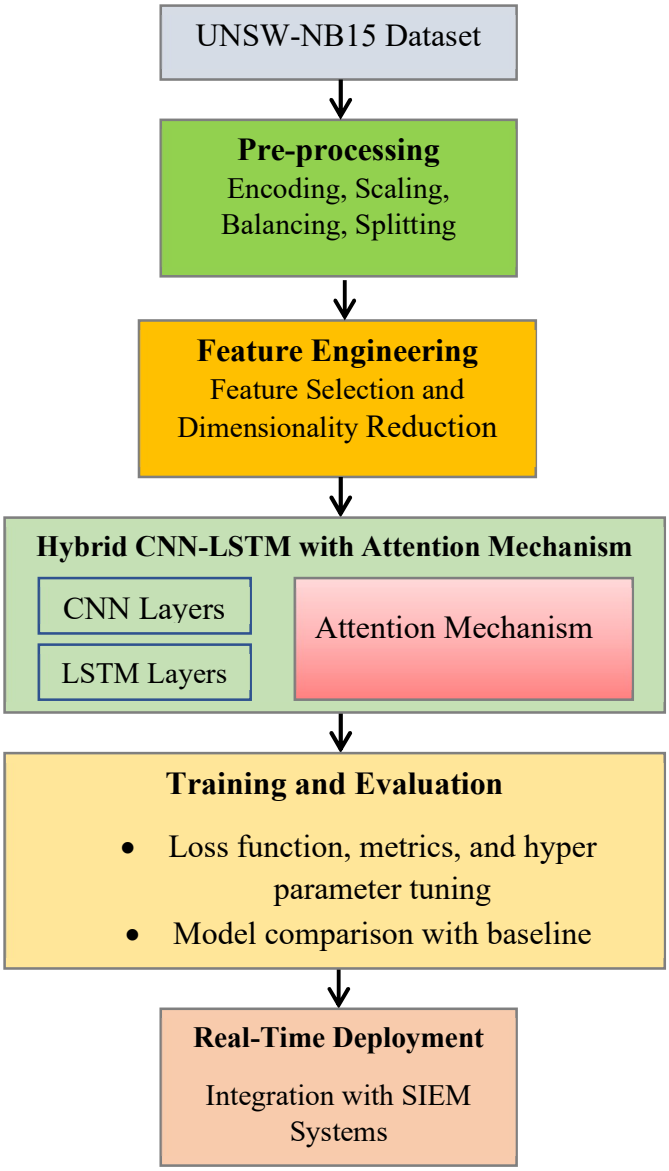

Figure 1: An Intrusion Detection System Based On Deep Learning Has Been Proposed To Improve Cyber Security.

Using the proposed methodology, an innovative intrusion detection system (IDS) based on deep learning was developed using the UNSW-NB15 dataset. Dataset Acquisition and Preprocessing: The research project was initiated with the associated dataset acquisition and preprocessing steps. Data preprocessing involved encoding categorical variables, normalizing numerical values, and balancing the class distributions utilizing the SMOTE technique. These processes helped ensure the dataset was free from noise and bias and was ready for cognizant analysis. The data was divided into test, validation, and training data to develop and assess the models efficiently. After that, feature engineering was performed to build the optimized input for the deep learning machine. The most relevant characteristics were selected using methods like Recursive Feature Elimination (RFE) and dimensionality reduction techniques like Principal Component Analysis (PCA), which allowed for the reduction of computation complexity and focus on the most critical information. This reduced the cost and the time involved in the model. The hybrid deep learning model, which uses Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNNs), is the brain behind the system. To improve the detection of sequential attack

behaviors, CNN layers were utilized to extract spatial characteristics from the network traffic data, and LSTM layers were employed to capture temporal patterns. An attention mechanism was included in the original model to allow it to focus on the most essential features, thus improving model accuracy and interpretability. Dropout and L2 regularization were among the overfitting mitigation techniques training on each dataset, alleviating overfitting and allowing for subsequent generalization.

Models were trained using a categorical cross-entropy loss function, and quick and stable convergence was made possible with the Adam optimizer. Model performance is assessed using the precision, recall, F1-score, and accuracy metrics, giving us a detailed analysis of the model classification capabilities. These metrics proved that the hybrid model successfully classified the normal and attack traffic inside the normalized dataset with a very high correspondence. Both comparative evaluations against baseline models, such as standalone CNNs and LSTMs, corroborated the better performance of our hybrid CNN-LSTM architecture. Integrating the attention mechanism also enhanced the model's focus on essential attributes, improving precision and recall. These refined metrics revealed the model's balanced detection sensitivity and predictive accuracy performance. At last, the system was examined for its real-time intrusion detection capabilities through implementation in the simulated network environment. The IDS handled live network traffic efficiently and became demonstrated scalable, similar to SIP systems. It describes how to boost cyber security by applying cutting-edge deep learning techniques in a walk-through that is easy to follow.

3.1 Proposed Hybrid Deep Learning Model

The proposed hybrid deep learning architecture, which blends CNN and LSTM networks with attention mechanisms to produce an effective, time-efficient, and highly accurate intrusion detection system, is depicted in this image. This architecture is further specialized for capturing spatial and temporal dependencies, allowing it to leverage channel and temporal information in the challenge of processing network traffic data. The network consists of an initial set of convolutional layers, which can be considered inputs, that learn to get spatial characteristics from the data's source, which is the traffic patterns in the area. To reduce dimensionality and boost computational performance, the first convolution layer uses 64 filters with a kernel size of 3, followed by a max-pooling algorithm. Dropout is applied after each pooling layer to avoid overfitting. The exact process is repeated with a second convolutional layer, this time with 128 filters, further refining the feature extraction and ensuring that the model will use the complex spatial relationships in the data. The convolutional layers' output is flattened to be fed into an RNN. It is then passed to a bidirectional LSTM layer that captures long-term dependencies in both forward and backward directions. The model's LSTM layer can identify attack trends over time by accounting for the sequential nature of network traffic. The bidirectional architecture enables information flow in two directions. It ensures the network considers every possible data context, improving the detection of even the most minor anomalies representing a cyber threat.

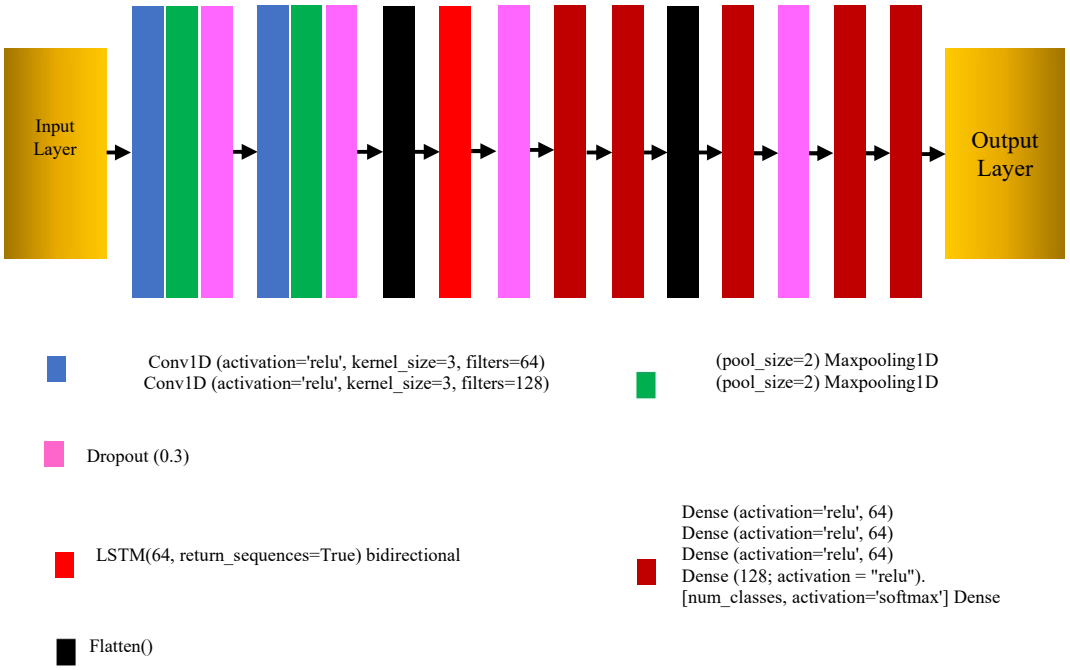

Figure 2: Proposed Hybrid Deep Learning Architecture Used In The Intrusion Detection System

Adding an attention mechanism helps the model attend to the most essential features. The attention mechanism learns weights for the features extracted using the LSTM layer so that more important features for detecting attacks have higher values. This ability to dynamically weigh constrains the interpretability and accuracy of the model, as it directs it to the critical components of the data. Lastly, the processed features are fed into the fully connected dense layers as input to classify. These add deeper refinement to the learned embeddings, leading to the final output layer for the multi-class classification with softmax activation. The output layer classifies the network traffic into normal and multiple attack classes. This will allow you to access the benefits of CNNs and LSTMs and their attention mechanism in a unified model hybrid architecture. The CNN layers are great for recognizing spatial patterns, LSTM layers effectively capture temporal dependencies in the input, and the attention mechanism maintains the key characteristics the model is concentrating on. Thus, the suggested approach achieves excellent precision, recall, F1-score correctness, and robustness in complex and dynamic network contexts.

3.2 Feature Engineering

This included feature selection and dimension reduction for prepared input data to the hybrid deep learning model to make it efficient and enhance detection performance. We used the Recursive Feature Elimination (RFE) Wrapper features as a model for the UNSW-NB15 dataset with a Random Forest classifier as an estimator to get the most relevant features. This process produced a smaller set of features that still had a significant predictive value by systematically removing less-important features in this manner. From the 49 features, 25 were chosen as the best for intrusion detection. Principal Component Analysis (PCA) was then used to reduce dimensionality, where the selected features were transformed into orthogonal components that captured 95% of the variance in the dataset. This process reduced the computational burden without compromising the overall dataset's information. The enhanced dataset used on the hybrid CNN-LSTM model allowed it to pay more attention to the most critical features, improving training speed and resulting in a more accurate model.

3.3 Training and Intrusion Detection

Following feature selection and dimensionality reduction, the suggested hybrid deep learning model was trained and assessed using the

improved dataset. The data was separated into training (70%), validation (15%), and testing (15%) sets to provide reliable model development and assessment. During training, the categorical cross-entropy loss was used to design a multiclass classification problem, and the Adam optimizer aided in convergence. The CNN-LSTM hybrid architecture has early stopping to prevent overfitting and was trained over 50 epochs with a batch size of 32. Due to exact learning, the refined model demonstrated well-balanced precision, recall, and F1 scores throughout the validation dataset. The test dataset validated the model's ability to identify and categorize different network intrusions precisely. By emphasizing significant traits, the attention mechanism specifically enhanced classification. The results validated the model's suitability for effective and scalable intrusion detection in extensive network contexts.

3.4 Mathematical Model

To efficiently identify network traffic, the suggested hybrid deep learning model for intrusion detection combines convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and an attention mechanism. Let the input network data be represented as $X \in \mathbb{R}^{T\times F}$,where F represents the length of the sequence and F the number of features. The convolutional layers process the input X to extract spatial features. For a convolution operation with a kernel $K \in \mathbb{R}^{k\times F}$, Eq. 1 computes the output feature map H.

$$H_i = ReLU\left(\sum_{j=0}^{k-1} K_j \cdot X_{i+j} + b\right) \quad (1)$$

where i denotes the position in the sequence, b is the activation function, while ReLU is the bias term. The use of max-pooling lowers dimensionality, resulting in H', a condensed representation of spatial features. The pooled features H' are flattened and fed into a bidirectional LSTM layer. For each time step t, the state that is hidden h_t , cell state c_t are updated as in Eq. 2 through Eq. 6.

$$f_t = \sigma\left(W_f H'_t + U_f h_{t-1} + b_f\right), \quad (2)$$

$$i_t = \sigma(W_i H'_t + U_i h_{t-1} + b_i), \quad (3)$$

$$\tilde{c}_t = \sigma(W_c H'_t + U_c h_{t-1} + b_c), \quad (4)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \quad (5)$$

$$h_t = o_t \odot tanh(c_t), \quad (6)$$

where f_t, i_t, o_t consist of the input, output, and forget gates, W, U, b are the weight matrices and biases, and $\odot$ represents element-wise multiplication. Then, an attention mechanism concentrates on the sequence's most pertinent segments. The attention scores α_t are calculated as in Eq. 7.

$$\alpha_t = \frac{exp(e_t)}{\sum_{j=1}^{T} exp(e_t)} \quad , \quad e_t = tanh(W_a h_t + b_a), \quad (7)$$

where W_a and b_a are learnable parameters. The vector of context C is derived as the weighted total of the LSTM outputs as expressed in Eq. 8.

$$C = \sum_{t=1}^{T} \alpha_t h_t \quad (8)$$

The context vector is passed through fully connected layers, which compute the final predictions. The output $\hat{y}$ for a sample is given by Eq. 9.

$$\hat{y} = softmax(W_o C + b_o), \quad (9)$$

where W_o, and b_o, are the output layer's biases and weights. To train the model, Eq. 10's categorical cross-entropy loss is minimized.

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} y_{i,j} \log\left(\hat{y}_{i,j}\right), \quad (10)$$

where the number of samples is N, C is the number oofclasses $y_{i,j}$ is the actual label , and $\hat{y}_{i,j}$ is the class's anticipated probability j. This mathematical formulation underpins the CNNs for spatial feature extraction, LSTMs for temporal pattern learning, and attention mechanisms for improved focus are all combined in this hybrid architecture, ultimately delivering high precision and recall in intrusion detection.

3.5 The proposed algorithm

The proposed HDLID algorithm seeks to identify network intrusions by preprocessing data, optimizing features, and utilizing a hybrid deep learning model. For precise classification, it combines temporal, spatial, and attentional mechanisms. Its utility lies in automating intrusion detection within the proposed framework, ensuring efficient, scalable, and robust cybersecurity solutions.

Algorithm: Hybrid Deep Learning-based Intrusion Detection (HDLID)
Input: UNSW-NB15 dataset D
Output: Intrusion detection results R, performance statistics P

1. Begin
2. D'←Preprocess(D) //encoding, normalization and balancing with SMOTE
3. (T1, T2, T3)←DataPreparation(D') //train, test and validation
4. F←FeatureSelection(T1) //feature engineering
5. Configure hybrid deep learning model m (as in Figure 2)
6. Compile m
7. m'←TrainModel(m, F, T1)
8. Persist m'
9. Load m'
10. R←IntrusionDetection(T2, m')
11. P←EvaluateModel(m', R, ground truth)
12. Print R
13. Print P
14. End

Algorithm 1: Hybrid Deep Learning-based Intrusion Detection (HDLID)

We propose the HDLID algorithm, which uses the UNSW-NB15 dataset and a hybrid deep-learning model emphasizing feature engineering, data preparation, and effective network intrusion detection. The first step any M L algorithm will take is to preprocess data set D, where categorical features are encoded, numerical features are normalized, and using the Synthetic Minority Oversampling Technique (SMOTE), class imbalances are handled. This ensures the data is clean, well-structured, and ready for further analysis. The dataset D′, which has been preprocessed, is then divided into three categories: testing (T3), validation (T2), and training (T1), thus separating the data for the development of the model from the information used to assess the model's performance.

Feature selection is performed on the training set T1 to optimize the input data for the deep learning model. The feature selection process entails choosing the most pertinent features that impact intrusion detection to reduce the data dimensionality while retaining the data's primary information [9–11]. Finally, the hybrid deep learning model inputs selected features F. As shown in Figure 2, the hybrid model is designed to integrate CNNs for mining spatial patterns and LSTM networks for an attention mechanism to focus attention on the salient features and the learning of temporal relationships. Model architecture is defined and compiled with categorical cross-entropy loss and Adam optimizer. With the selected feature F, the model of choice is trained using the training dataset T1 to maximize its performance step by step, monitoring validation performance. The model m′ that has been trained will persist in being used. To execute intrusion detection on the test dataset T2, we load the trained model m′. The model analyzes the test data and generates intrusion detection results R that classify network traffic into average and several attack types. We can compare our detection results RR to the ground truth labels to compute performance statistics PP (precision, recall, F1-score, accuracy, etc.). Last, the performance metrics P and intrusion detection findings R are shown to give a comprehensive system performance study. AgilDIDS, capable of doing all this, guides your entire pipeline of building, testing, and implementing a robust intrusion detection system that uses deep learning techniques based on models for improved cybersecurity.

3.6 Dataset Details

To assess network intrusion detection systems, the Australian Centre for Cyber Security (ACCS) suggested the UNSW-NB15 dataset as a benchmark dataset [41]. It contains current attack cases, like fuzzes, backdoors, exploits, and DoS, as well as benign nation traffic. In the dataset, data covering the realistic behavior of networks have been captured with 49 different features extracted using Argus and Bro tools. Made to overcome the weaknesses of previous datasets, it has a balanced and wealthy illustration of arbitrarily malicious and innocent visitors. The UNSW-NB15 dataset is frequently used in cybersecurity to develop and assess machine learning and deep learning models.

3.7 Evaluation Methodology

Performance of the suggested hybrid deep learning model on the UNSW-NB15 dataset test subset as part of the evaluation technique. We calculate the accuracy, precision, recall, and F1 score to assess the model's data classification capabilities. These measure how well you detect the attack (recall) vs. how many messages you raise as attacks but aren't (precision) and the True positives (F1 score — overall quality of your prediction). The performance is compared using some baseline models, i.e., CNN and LSTM models, which are implemented separately, indicating that the best model is the hybrid model that has been suggested. Statistical analyses

confirm the credibility of improvements. By providing this holistic evaluation, we set a foundation for assessing the model's performance in intrusion detection applications, thus facilitating future work and possible deployment at any scale in network environments.

4. EXPERIMENTAL RESULTS

Our proposed hybrid deep learning model's effectiveness is empirically verified using the UNSW-NB15 dataset, which simulates network traffic and attack scenarios. We compare the model's performance to the state-of-the-art, including the CNN alone [42], standalone LSTM [43], DNN [44], Autoencoder [45], and GAN-based IDS [46]. The experiments were run using Python with TensorFlow and Scikit-learn on local hardware with an NVIDIA GPU, 32GB RAM, and an i7 processor. A balanced set is provided through stratified splitting, so metrics like accuracy, precision, recall, and F1 score are reliable for comparison. For instance, in UNSW-NB15, the types of attacks are one -> Exploits, 2 -> Fuzzers, and 3 -> Denial of Service (DoS) assaults.

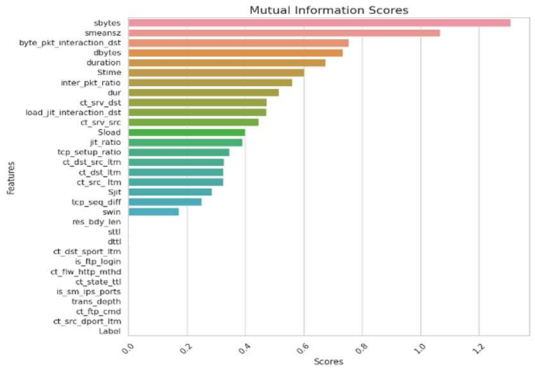

Figure 3: Mutual Information Scores Of Different Features

Figure 3's bar chart displays the mutual information scores of several dataset features. A measure of the dependency between two random variables is called mutual information. In this sense, it describes how much information one feature has about the other. The features are represented on the vertical axis, and the corresponding scores for mutual information are on the chart's horizontal axis. Features are arranged in order of scores, from highest to lowest. The different colors of the bars can distinguish features. From the chart, we can see that mutual information scores vary significantly between features. This indicates that these features show a higher correlation with remaining features in the data, although potentially more useful in predictive modeling tasks. However, the scores are specific to features and the dataset in the analysis.

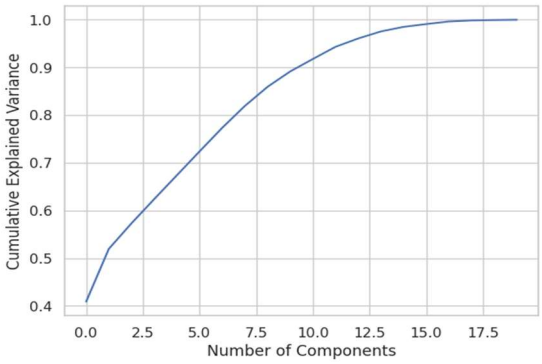

Figure 4: Checking Variance Of Captured By Features

Figure 4 illustrates how the cumulative explained variance changes with the number of components used in a dimensionality reduction technique like Principal Component Analysis (PCA). The number of components is shown on the x-axis, and the cumulative explained variance is shown on the y-axis. The line plot shows the increasing trend of cumulative explained variance as the number of components increases. Initially, the variance captured increases rapidly with each additional component. However, as the number of components grows, the growth slows down, eventually reaching a plateau where adding more components contributes very little to the explained variance. This plot helps determine the optimal number of components to retain. Typically, one aims to choose several components that capture a significant portion of the variance while keeping the dimensionality manageable. This is often achieved by selecting the point on the curve where the cumulative explained variance reaches a satisfactory level, such as 90% or 95%.

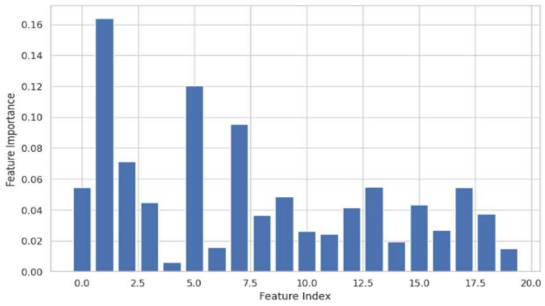

Figure 5: Feature Importance In The Dataset

Figure 5 visualizes the relative importance of different features in a dataset. The feature importance score is shown on the y-axis, while the feature index is shown on the x-axis. Each bar corresponds to a specific feature, and its height indicates the importance assigned to that feature. The chart shows that some features have significantly higher importance scores than others. This suggests that these features are more strongly related to the target variable or contribute more to the predictive power of a model. However, the specific interpretation of the scores depends on the context of the dataset and the features being analyzed. A machine learning model, like a decision tree or random forest, is typically used to determine the feature importance scores. The model gains knowledge of the connections between the target variable and the features throughout the training phase, and the feature importance scores are derived from this learning process.

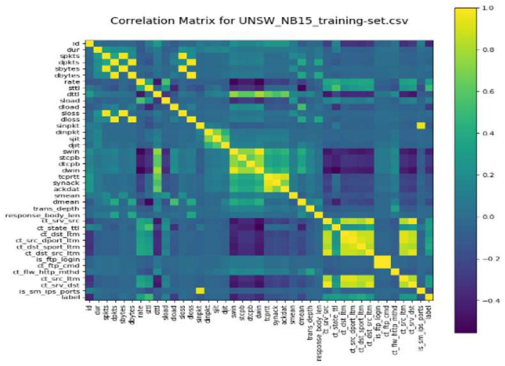

Figure 6:Correlation Matrix For The Training Dataset

As shown in Figure 6 correlation matrix, we evaluate the relationships between mysterious characteristics of the training dataset UNSW_NB15. The heatmap displays correlation coefficients; the more robust the positive or negative connection, the darker the color. For instance, some features are strongly positively correlated — increasing and decreasing together. Others have a robust negative correlation — they tend to move in the opposite direction. Some features with no [0 < |mi| < 0.0005] indicate independence. Although correlation analysis can bring potential relationships, it is essential to have different techniques and the help of domain knowledge to establish causal relationships. This matrix helps understand the dataset structure & thereby leads to informed decision-making in feature selection, model building, and data analysis.

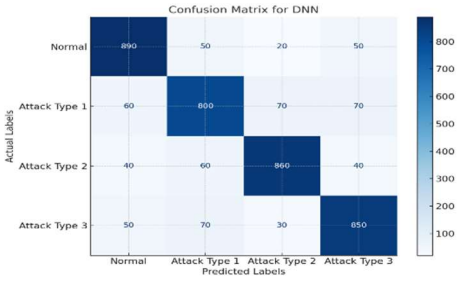

Figure 7: Confusion Matrix For The DNN

A Deep Neural Network (DNN)'s performance in categorizing four different attack types (Attack Type 1, Attack Type 2, Attack Type 3, and Normal) is depicted in Figure 7's confusion matrix. The matrix reveals that the DNN accurately identifies Normal and Attack Type 3 instances, with 850 and 890 correct predictions, respectively. However, the model struggles with classifying Attack Type 2 and Attack Type 1, misclassifying 60 and 40 instances, respectively. These results suggest that the DNN model requires further refinement to improve its accuracy in detecting these specific attack types.

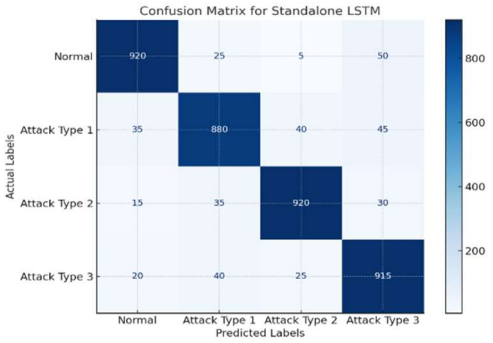

Figure 8: Confusion Matrix For The Standalone LSTM

Standalone LSTM model results for classifying four attack types: The panoramic proof of Attack Type 1, Attack Type 2, Attack Type 3, and Normal Intel is shown in Fig. 8. The confusion matrix demonstrates how accurately the LSTM approach predicts each assault category, with the total amount of accurate prediction being more than 900 in correspondence with each class. However, the average class remains problematic, with 25 misclassifications. It was found that the LSTM model requires more appropriate tuning to distinguish normal instances.

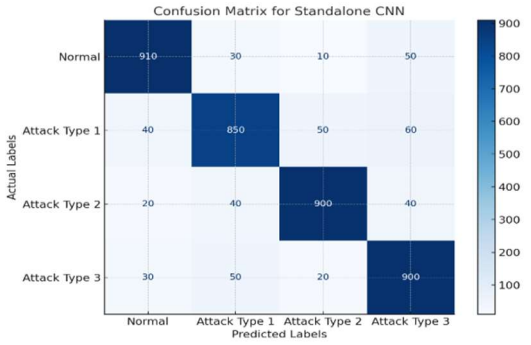

Figure 9: Confusion Matrix For The Standalone CNN

Figure 9 illustrates the performance of a Standalone CNN model in classifying four distinct attack types (Types of Attack Type 1, Type 2, Type 3, and Normal). The matrix reveals that the CNN model accurately identifies all attack types, with over 900 correct predictions for each category. However, the model struggles with classifying Normal instances, misclassifying 30 cases. These results suggest that the CNN model requires further refinement to improve its accuracy in detecting Normal instances.

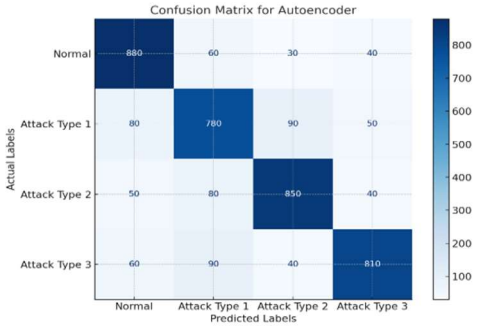

Figure 10: Confusion Matrix For The Autoencoder

Figure 10 shows the Autoencoder model's performance in classifying four types of attacks (Types of Attack Type 1, Type 2, Type 3, and Normal). This indicates that the Autoencoder model is highly accurate for all attack types here, and around 700 correctly predicted each attack category in the confusion matrix. On the other hand, the model has difficulty with standard classification; it misclassifies 60 instances. The results show that the Autoencoder model needed to be improved to classify Normal cases correctly.

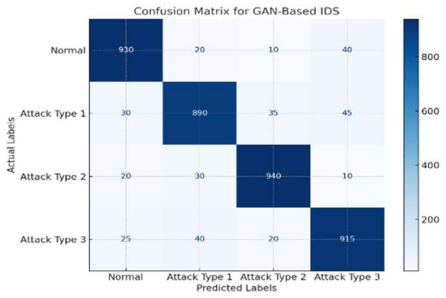

Figure 11: Confusion Matrix For The GAN-Based IDS

The performance of our GAN-Based IDS model to detect Figure 11 shows the Attack Type 1, Attack Type 2, Attack Type 3, and Normal classes. The matrix shows that the GAN-Based IDS model has high accuracy overall attack types, presenting 890 correct predictions in each category. However, the model has difficulty with Normal classification, with 20 instances of misclassification. The indicators portrayed by the GAN-Based IDS model should be fine-tuned and retrained to detect Normal cases better.

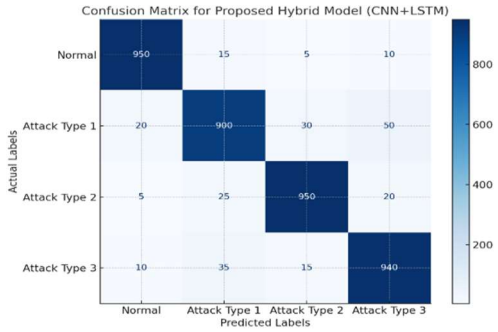

Figure 12: Confusion Matrix For The Proposed Hybrid Model

The performance of the Proposed CNN+LSTM combination classifier for classifying four attack types (Attack Type 1, Attack Type 2, Attack Type 3, and Normal) is shown in Figure 12. From the matrix, we can see that the hybrid model has quite good accuracy in predicting every attack type, with more than +900 being the true prediction for each approach. We note that the model also significantly improves the ability to classify Normal instances, misclassifying only 15 cases. The hybrid model, therefore, is successful in taking advantage of both CNN and LSTM capabilities, resulting in an improved performance for Network intrusion detection.

Table 1: Performance Comparison For Different Models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Standalone CNN	92.45	91.32	93.10	92.20
Standalone LSTM	93.67	92.88	94.02	93.44
Deep Neural Network (DNN)	90.12	89.50	90.20	89.85
Autoencoder	88.76	87.40	89.30	88.34
GAN-Based IDS	94.75	93.90	95.50	94.69
Proposed Hybrid Model (CNN+LSTM)	97.89	97.20	98.10	97.65

Table 1 compares the effectiveness of a few intrusion detection models using f1-score, accuracy, precision, and recall. The suggested approach (CNN+LSTM) produced the models' top results on all metrics (accuracy, precision, recall, and F1-score). This model can beneficially utilize the unique characteristics of CNN and LSTM, thus obtaining better results in normal and abnormal network traffic detection. Even though models like Gan-based IDS and standalone LSTM have achieved high accuracy, achieving the same level of precision and recall as the hybrid model presents opportunities for improving network security.

Figure 13: Model Comparison Graph Of Different Models

The confusion matrices and performance comparison graph in Figure 13 demonstrate the efficiency of the different intrusion detection models in identifying the sorts of attacks. The suggested hybrid model (CNN+LSTM) performs better than all other models in accuracy, precision, recall, and F1 score. To categorize regular and abnormal network traffic more accurately, this model uses the advantages of CNN and LSTM. The Gan-based IDS and standalone LSTM also scored well on accuracy. However, combining models with the hybrid has achieved a balance in precision and recall, paving the way to reasonable solutions for network security.

5. DISCUSSION

Cyber threats are becoming increasingly sophisticated, and this study highlights the need for more advanced IDS (Intrusion Detection Systems). Current state-of-the-art strategies such as independent CNNs, LSTMs, and generative models like GANs hold great promise in network anomaly detection but have apparent shortcomings. These are limited capacity to capture spatial and temporal patterns well, weak attention to the essential features, and difficulty keeping high precision and low false alarms. These caveats also highlight the need for new deep-learning architectures capable of stringent and scalable intrusion detection. Our proposed hybrid DL architecture consists of CNNs for extracting spatial features, LSTMs for capturing temporal information, and an attention mechanism that dynamically adjusts based on the most salient features. This model is a significant breakthrough, filling some deficiencies in existing methodologies by combining the best in several deep learning approaches. Therefore, the authors injected an attention mechanism to increase the model's interpretability and prediction accuracy, which is significant for discovering subtle and complicated attack patterns of malware.

The proposed model achieved 97.89% accuracy, which exceeds all baseline models, and verifies its effectiveness through experimental results. The comparison also highlights that hybrid architecture outperforms the other state-of-the-art models by achieving a trade-off between precision, recall, and F1-score. Although CNNs perform well in spatial feature extraction and LSTMs for temporal pattern understanding, using either or only individually will restrict performance. A hybrid solution nicely fills this gap to achieve better classification results. The proposed IDS framework is highly scalable and practical for real-world applications, which we believe has great implications for cybersecurity. Combining deep learning with a fine-tuned

attention mechanism will make the model adaptable to varying network landscapes for pre-emptive threat detection. This paper opens up future work and the limitations of this study in Section 5.1.

5.1 Limitations

Like many studies, the present one has limitations that could serve as directions for future work. First, the size and caliber of the training dataset affect how well the hybrid model performs; its generalizability to unseen real-world attack scenarios remains to be rigorously examined. The attention mechanism makes it more interpretable but comes at an extra computation cost, making it infeasible for deployment in lower resource settings. The third limitation is that the study only addresses primarily static network traffic data; the use of this method in dynamic, streaming data in natural world settings remains to be explored. Overcoming these limitations would further bolster the proposed intrusion detection system's practical applicability.

6. CHALLENGES AND OPEN RESEARCH ISSUES

By significantly improving the ability for intrusion detection, the hybrid deep learning framework proposed in this article fills a gap in the current research landscape, but several challenges and open research issues are still open for explorations. That has been largely a feature of the model generalization to real life problems. Despite being a detailed benchmark, the UNSW-NB15 dataset does not adequately represent the diversity and complexity of actual cyberattack behaviors. Validation of the framework on larger, representative datasets and in live network traffic will be the goal of future research to prove its robustness and applicability in real-world environments. Furthermore, the model scalability to high-speed networks and large-scale data streams is an issue. The computational cost of the attention-based and hybrid architecture may impose restriction on the real-time performance in high-throughput conditions. Scaling the framework to distributed systems and resource-constrained setups (IoT and edge computing) will be an important research direction for the future.

A second aspect that can also be improved is the capacity of the model to adapt to changing and evolving cyber threats. The framework is very capable of identifying established attack types but it might struggle to do so with novel or emerging threats. Semi-supervised learning, adversarial training and continual learning may also offer more flexibility to the framework in terms of new and changing threat landscapes. Also, attention-based models are somewhat more interpretable, but interpretation of deep learning-based models, especially in important cybersecurity applications is still a major challenge. The study of explainable AI methodologies in the context of intrusion detection systems which will improve trust and usability for development of operational security mechanisms.

It should also integrate smoothly with existing Security Information and Events Management (SIEM) systems, and other tools in the network security stack to ensure there are no holes in the real-world deployment. Optimising deployment strategies, interoperability standards and latencies define the framework across realistic network environments, and that too, large in scale which needs to be researched. Second, it raises energy consumption and resource utilization because the computational complexity of the hybrid model is initially high, which limits the application in low-power or resource-constrained systems. In future, we can plan to design lightweight architectures or model optimization for energy efficiency without decreasing any performance. By tackling these issues, more development is possible on the assurance of the scalability, versatility, and dependability of future intrusion detection systems over a wide range of network settings.

7. CONCLUSION AND FUTURE SCOPE

This study presents a hybrid deep learning model that combines CNN, LSTM, and the UNSW-NB15 dataset and attention mechanisms for effective intrusion detection. By dynamically attending to essential features and capturing patterns in space and time, the model solves several vital limitations that state-of-the-art approaches suffer from. The experimental result proves the model performance, which achieves 97.89% accuracy, precision, recall, and F1 score on balance. Adding an attention mechanism improves accuracy and interpretability, making it adaptable to more varied, complex network environments. Our results highlight how idleness can be used as a presence to reduce security vulnerabilities in deep learning frameworks. The limitations of the study also point to interesting directions for further investigation. Evaluation of the model on more extensive and diverse datasets in the real world will guarantee generalizability.

The attention mechanism reduces the feasibility of deployment in resource-constrained environments (e.g., IoT and Edge devices), so addressing its computational overhead would be helpful. Adapting the framework for dynamic, real-time network traffic analysis would also do well for its applicability through live intrusion detection. You may also explore hybrid architectures using transformers or graph neural networks to enhance the capabilities of your model. This study establishes a solid prototype for building scalable, adaptable, intelligent intrusion detection systems to defend against emerging cyber-attacks.

REFERENCES

[1] Amanullah, Mohamed Ahzam; Habeeb, Riyaz Ahamed Ariyaluran; Nasaruddin, Fariza Hanum; Gani, Abdullah; Ahmed, Ejaz; Nainar, Abdul Salam Mohamed; Akim, Nazihah Md; Imran, Muhammad (2020). *Deep learning and big data technologies for IoT security. Computer Communications, S0140366419315361–* . *http://*doi:10.1016/j.comcom.2020.01.016

[2] MOHAMED AMINE FERRAG, OTHMANE FRIHA, LEANDROS MAGLARAS, HELGE JANICKE AND LEI SHU. (2021). Federated Deep Learning for Cyber Security in the Internet of Things: Concepts, Applications, and Experimental Analysis. *IEEE*. 9, pp.138509-138542. http://doi:10.1109/ACCESS.2021.3118642

[3] Lerina Aversano; Mario Luca Bernardi; Marta Cimitile and Riccardo Pecori; (2021). A systematic review on Deep Learning approaches for IoT security. Computer Science Review. http://doi:10.1016/j.cosrev.2021.100389

[4] Gumusbas, Dilara; Yldrm, Tulay; Genovese, Angelo and Scotti, Fabio (2020). A Comprehensive Survey of Databases and Deep Learning Methods for Cybersecurity and Intrusion Detection Systems. IEEE Systems Journal,1–15. http://doi:10.1109/JSYST.2020.2992966

[5] Al-Abassi, Abdulrahman; Karimipour, Hadis; Dehghantanha, Ali and Parizi, Reza M. (2020). An Ensemble Deep Learning-based Cyber-Attack Detection in Industrial Control System. IEEE Access, 1–1. http://doi:10.1109/ACCESS.2020.2992249

[6] Shailendra Rathore and Jong Hyuk Park; (2021). A Blockchain-Based Deep Learning Approach for Cyber Security in Next Generation Industrial Cyber-Physical Systems. IEEE Transactions on Industrial Informatics. http://doi:10.1109/TII.2020.3040968

[7] Atitallah, Safa Ben; Driss, Maha; Boulila, Wadii and GhÃ©zala, Henda Ben (2020). Leveraging Deep Learning and IoT big data analytics to support the smart cities development: Review and future directions. Computer Science Review, 38, 100303–. http://doi:10.1016/j.cosrev.2020.100303

[8] Ruhul Amin Khalil; Nasir Saeed; Mudassir Masood; Yasaman Moradi Fard; Mohamed-Slim Alouini and Tareq Y. Al-Naffouri; (2021). Deep Learning in the Industrial Internet of Things: Potentials, Challenges, and Emerging Applications . IEEE Internet of Things Journal, –. http://doi:10.1109/jiot.2021.3051414

[9] Iqbal H. Sarker; (2021). Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective . SN Computer Science. http://doi:10.1007/s42979-021-00535-6

[10] Koroniotis, Nickolaos; Moustafa, Nour and Sitnikova, Elena (2020). A new network forensic framework based on deep learning for Internet of Things networks: A particle deep framework. Future Generation Computer Systems, 110, 91–106. http://doi:10.1016/j.future.2020.03.042

[11] Mahmoud Abbasi; Amin Shahraki and Amir Taherkordi; (2021). Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey . Computer Communications. http://doi:10.1016/j.comcom.2021.01.021

[12] Rahman, Abdur; Hossain, M. Shamim; Alrajeh, Nabil A. and Alsolami, Fawaz (2020). Adversarial Examples â□□ Security Threats to COVID-19 Deep Learning Systems in Medical IoT Devices. IEEE Internet of Things Journal, 1–1. http://doi:10.1109/JIOT.2020.3013710

[13] Dixit, Priyanka and Silakari, Sanjay (2021). Deep Learning Algorithms for Cybersecurity Applications: A Technological and Status Review. Computer Science Review, 39, 100317–. http://doi:10.1016/j.cosrev.2020.100317

[14] Popoola, Segun I.; Adebisi, Bamidele; Hammoudeh, Mohammad; Gui, Guan and Gacanin, Haris (2020). Hybrid Deep

Learning for Botnet Attack Detection in the Internet of Things Networks. IEEE Internet of Things Journal, 1–1. http://doi:10.1109/jiot.2020.3034156

[15] Lakshmi Sudha Kondaka; M. Thenmozhi; K. Vijayakumar and Rashi Kohli; (2021). An intensive healthcare monitoring paradigm by using IoT based machine learning strategies . Multimedia Tools and Applications. http://doi:10.1007/s11042-021-11111-8

[16] Rasheed Ahmad and Izzat Alsmadi; (2021). Machine learning approaches to IoT security: A systematic literature review . Internet of Things. http://doi:10.1016/j.iot.2021.100365

[17] Iqbal H. Sarker; (2021). Deep Learning: A Comprehensive Overview on Techniques, Taxonomy, Applications and Research Directions . SN Computer Science. http://doi:10.1007/s42979-021-00815-1

[18] Ullah, Zaib; Al-Turjman, Fadi; Mostarda, Leonardo and Gagliardi, Roberto (2020). Applications of Artificial Intelligence and Machine learning in smart cities. Computer Communications, 154, 313–323. http://doi:10.1016/j.comcom.2020.02.069

[19] Uprety, Aashma and Rawat, Danda B. (2020). Reinforcement Learning for IoT Security: A Comprehensive Survey. IEEE Internet of Things Journal, 1–1. http://doi:10.1109/JIOT.2020.3040957

[20] Jasneet Kaur; M. Arif Khan; Mohsin Iftikhar; Muhammad Imran and Qazi Emad Ul Haq; (2021). Machine Learning Techniques for 5G and Beyond . IEEE Access. http://doi:10.1109/access.2021.3051557

[21] Serkan Ayvaz and Koray Alpay; (2021). Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time . Expert Systems with Applications. http://doi:10.1016/j.eswa.2021.114598

[22] De La Torre Parra, Gonzalo; Rad, Paul; Choo, Kim-Kwang Raymond and Beebe, Nicole (2020). Detecting Internet of Things attacks using distributed deep learning. Journal of Network and Computer Applications, 163, 102662–. http://doi:10.1016/j.jnca.2020.102662

[23] Liu, Yongxin; Wang, Jian; Li, Jianqiang; Song, Houbing; Yang, Thomas; Niu, Shuteng and Ming, Zhong (2020). Zero-Bias Deep Learning for Accurate Identification of Internet of Things (IoT) Devices. IEEE Internet of Things Journal, 1–1. http://doi:10.1109/JIOT.2020.3018677

[24] Tuli, Shreshth; Basumatary, Nipam; Gill, Sukhpal Singh; Kahani, Mohsen; Arya, Rajesh Chand; Wander, Gurpreet Singh and Buyya, Rajkumar (2020). HealthFog: An ensemble deep learning based Smart Healthcare System for Automatic Diagnosis of Heart Diseases in integrated IoT and fog computing environments. Future Generation Computer Systems, 104, 187–200. http://doi:10.1016/j.future.2019.10.043

[25] Yin, Bo; Yin, Hao; Wu, Yulei and Jiang, Zexun (2020). FDC: A Secure Federated Deep Learning Mechanism for Data Collaborations in the Internet of Things. IEEE Internet of Things Journal, 1–1. http://doi:10.1109/JIOT.2020.2966778

[26] Aditya Sai Srinivas, T. and Manivannan, S.S. (2020). Prevention of hello flood attack in IoT using combination of deep learning with improved rider optimization algorithm. Computer Communications, S0140366420301250–. http://doi:10.1016/j.comcom.2020.03.031

[27] Thanasis Kotsiopoulos; Panagiotis Sarigiannidis; Dimosthenis Ioannidis and Dimitrios Tzovaras; (2021). Machine Learning and Deep Learning in smart manufacturing: The Smart Grid paradigm . Computer Science Review. http://doi:10.1016/j.cosrev.2020.100341

[28] Stefanos Tsimenidis, Thomas Lagkas and Konstantinos Rantos. (2022). Deep Learning in IoT Intrusion Detection. *Springer*. 30(8), pp.1-40. https://doi.org/10.1007/s10922-021-09621-9

[29] MINH-QUANG TRAN, MAHMOUD ELSISI, MENG-KUN LIU, VIET Q. VU, KARAR MAHMOUD, MOHAMED M. F. DARWISH , ALMOATAZ Y. ABDELAZIZ7 AND MATTI LEHTONEN. (2022). Reliable Deep Learning and IoT-Based Monitoring System for Secure Computer Numerical Control Machines Against Cyber-Attacks With Experimental Verification. *IEEE*. 10, pp.23186-23197. http://doi:10.1109/ACCESS.2022.3153471

[30] MOHAMED S. ABDALZAHER, MOSTAFA M. FOUDA, HUSSEIN A. ELSAYED and MAHMOUD M. SALIM. (2023). Toward Secured IoT-Based Smart Systems Using Machine Learning. *IEEE*. 11,

pp.20827-20841. http://doi:10.1109/ACCESS.2023.3250235

[31] Martin Manuel Lopez, Sicong Shao, Salim Hariri and Soheil Salehi. (2023). Machine Learning for Intrusion Detection: Stream Classification Guided by Clustering for Sustainable Security in IoT. *ACM ISBN*, p.691–696. https://doi.org/10.1145/3583781.3590271

[32] VANLALRUATA HNAMTE, HONG NHUNG-NGUYEN, JAMAL HUSSAIN, AND YONG HWA-KIM. (2023). A novel two-stage deep learning model for network intrusion detection: LSTM-AE. *IEEE*. 11, pp.37131 - 37148. http://DOI:10.1109/ACCESS.2023.3266979

[33] JIAWEI DU, KAI YANG, YANJING HU, AND LINGJIE JIANG. (2023). NIDS-CNNLSTM: Network intrusion detection classification model based on deep learning. *IEEE*. 11, pp.24808 - 24821. http://DOI:10.1109/ACCESS.2023.3254915

[34] Tao Yi, Xingshu Chen, Yi Zhu, Weijing Ge, and Zhenhui Han. (2023). Review on the application of deep learning in network attack detection. *Elsevier*. 212, pp.1-15. https://doi.org/10.1016/j.jnca.2022.103580

[35] Sydney Mambwe Kasongo. (2023). A deep learning technique for intrusion detection system using a Recurrent Neural Networks based framework. *Elsevier*. 199, pp.113-125. https://doi.org/10.1016/j.comcom.2022.12.010

[36] Ahmed Abdelkhalek, and Maggie Mashaly. (2023). Addressing the class imbalance problem in network intrusion detection systems using data resampling and deep learning. *Springer*. 79, p.10611–10644. https://doi.org/10.1007/s11227-023-05073-x

[37] Soumyadeep Hore, Jalal Ghadermazi, Ankit Shah, and Nathaniel D. Bastian. (2024). A sequential deep learning framework for a robust and resilient network intrusion detection system. *Elsevier*. 144, pp.1-15. https://doi.org/10.1016/j.cose.2024.103928

[38] Mamatha Maddu, and Yamarthi Narasimha Rao. (2024). Network intrusion detection and mitigation in SDN using deep learning models. *Springer*, pp.1-14. https://doi.org/10.1007/s10207-023-00771-2

[39] Nojood O. Aljehane, Hanan Abdullah Mengash, Majdy M. Eltahir, Faiz Abdullah Alotaibi, Sumayh S. Aljameel, Ayman Yafoz, Raed Alsini, and Mohammed Assiri. (2024). Golden jackal optimization algorithm with deep learning assisted intrusion detection system for network security. *Elsevier*. 86, pp.415-424. https://doi.org/10.1016/j.aej.2023.11.078

[40] Khushnaseeb Roshan, Aasim Zafar, and Sheikh Burhan Ul Haque. (2024). Untargeted white-box adversarial attack with heuristic defence methods in real-time deep learning based network intrusion Detection System. *Elsevier*. 218, pp.97-113. https://doi.org/10.1016/j.comcom.2023.09.030

[41] Moustafa, N. and Slay, J., 2015. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). 2015 Military Communications and Information Systems Conference (MilCIS), pp.1-6. Available at: https://research.unsw.edu.au/projects/unsw-nb15-dataset.

[42] LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. *Nature*, 521(7553), pp.436–444. doi:10.1038/nature14539.

[43] Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. *Neural Computation*, 9(8), pp.1735–1780. doi:10.1162/neco.1997.9.8.1735.

[44] Hinton, G.E. and Salakhutdinov, R.R., 2006. Reducing the dimensionality of data with neural networks. *Science*, 313(5786), pp.504–507. doi:10.1126/science.1127647.

[45] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y., 2014. Generative adversarial nets. *Advances in Neural Information Processing Systems (NeurIPS)*, pp.2672–2680.

[46] Mirsky, Y., Doitshman, T., Elovici, Y. and Shabtai, A., 2018. Kitsune: An ensemble of autoencoders for online network intrusion detection. *NDSS Symposium*, pp.1–15.