Improving Intrusion Detecction Systems by using Deep Learning Methods on Time Series Data

Asma Ahmed A. Mohammed
Department of Computer Science, University of Tabuk, Tabuk, Saudi Arabia
a.amohammed@ut.edu.sa (corresponding author)

Received: 26 October 2024 | Revised: 18 November 2024 | Accepted: 23 November 2024

 DOI: https://doi.org/10.48084/etasr.9417

ABSTRACT

Intrusion Detection Systems (IDSs) are the cornerstone of cybersecurity, monitoring network traffic to find abnormal suspicious activities. Traditional IDSs usually face challenges in adapting to the cyber threats that evolve day by day, leading to very high false positive rates and missed detections. This study focuses on enhancing the performance of an IDS system by integrating deep learning techniques with time series data. The efficiency of RNN, CNN, and LSTM networks was evaluated in detecting intrusions in real-time. The experimental results showed that hybrid models, especially the CNN+RNN+LSTM combination, performed best with a 0.86 F1 score, 0.92 precision, and 0.79 recall, indicating that hybrid deep learning methods can improve detection accuracy while reducing false alarms, opening a resilient future for cybersecurity.

Keywords-intrusion detection systems; deep learning; time series data; recurrent neural networks; long short-term memory; convolutional neural networks; cybersecurity

I. INTRODUCTION

ICT systems and networks form the backbone of today's digital world, in which information sharing and collaboration are comfortably extended to multiple platforms. However, all this interconnectedness shares sensitive user data, making them vulnerable to various types of cyber threats, both internally and externally. Cyberattacks have become increasingly sophisticated, breaching any security measures and then exploiting vulnerabilities [1]. For example, the Equifax data breach, which drew highly publicized attention, compromised personal data from more than 147 million individuals and caused immense financial losses and reputational damage, raising the dire need for effective intrusion detection mechanisms [2]. With the ever-evolving threat landscape, where attack methods are constantly updated, traditional cybersecurity measures have not met demands. This puts into motion the importance of developing high-end, adaptive, and reliable IDSs with capabilities for real-time threat identification and mitigation. An IDS is an active defense mechanism that involves continuously monitoring network traffic to detect and classify possible security breaches and flags off malicious activity based on predefined criteria, usually classified into two main classes: Network-based IDSs (NIDS) and host-based IDSs (HIDS) [3].

An NIDS focuses on network traffic data, obtained using various network devices, such as routers and switches, by leveraging techniques that scan packet contents for suspicious patterns and threats. On the other hand, a HIDS performs an analysis of the logs collected from specific hosts, depending on local sensors to identify bad behavior. This can be achieved using various types of log files, such as system and application logs [4]. Many organizations use hybrid approaches, benefiting from the two worlds of NIDS and HIDS to strengthen their security stance. Traditional IDS methods for traffic analysis often include misuse detection, anomaly detection, and stateful protocol analysis. Misuse detection relies on predefined signatures and filters for the detection of known threats, while anomaly detection uses several heuristic approaches that can uncover new attacks. This is invariably associated with a higher false alarm rate [5]. Stateful protocol analysis moves one step beyond, monitoring protocol behaviors over successive layers in its effort to identify a deviation from normal operation patterns. Despite the recent exponential use of ML techniques in the development of effective IDSs, the challenges remain formidable. Among them, the key issues are high false positive rates, limited generalizability by relying on single datasets, and weaknesses in addressing the scale and dynamics present in modern network traffic.

This study attempts to handle these challenges by assessing the efficiency of deep learning approaches applied to time series data-driven IDSs [6]. To improve detection accuracy and reduce false positive rates, several architectures were examined, namely CNN, RNN, and LSTM networks, and different attention mechanisms. The purpose of this study includes:

- Investigate the use of deep learning techniques to improve IDS capabilities.

- Evaluate the performance of deep learning models using time series data for intrusion detection.
- Determine how different architectures influence the detection accuracy and false-positive rates.

II. RELATED WORKS

In recent years, intrusion detection has undergone enormous variation, especially with the arrival of machine learning and deep learning techniques. In this regard, a relevant review showed that existing approaches are divided into two main classes, traditional methods and modern deep learning frameworks, with respective strengths and weaknesses [1, 7, 8].

A. Traditional Intrusion Detection Techniques

Traditionally, IDSs were highly reliant on signature-based detection methods. These systems use predefined rules and signatures extracted from previously known attack patterns to identify malicious activities. Although efficient against already known threats, signature-based systems lack efficiency in identifying new and unknown attacks, leaving significant vulnerabilities. As a complementary solution, anomaly detection has focused on finding deviations from established baselines of normal behavior. This is developed based on statistical techniques and algorithms in machine learning that identify unusual patterns featuring possible intrusions. However, traditional methods for anomaly detection usually bring a high number of false positives, as they rely on general patterns of behavior that, under some circumstances, may be mistakenly perceived as malicious. For example, an alert can be fired in a sudden network peak that reflects a very valid increase in activity [9]. In addition, the complexity of modern networks requires more sophisticated methods that can adapt to emerging attack vectors and traffic patterns.

B. Machine Learning Approaches

In recent times, machine learning has offered unparalleled capabilities to IDSs. Decision Trees (DT), SVM, and Random Forests (RF) have been used along with other ML algorithms for intrusion detection tasks. Previous studies have shown that ML models can classify network traffic into benign/malicious classes with improved detection rates for known attacks [10]. Some drawbacks include the reliance on static feature sets that hinder generalization across different datasets. Moreover, typical ML models lack the strength to learn temporal relationships, a prevalent aspect of network traffic data, and thus either miss detections or respond to an attack at a later stage.

C. Deep Learning Techniques

Recent advances in deep learning have unlocked unprecedented opportunities for enhancing IDS performance. Deep learning architectures, such as CNNs and LSTMs, can learn features hierarchically from raw data with little feature engineering involved in the process [7]. For instance, CNNs have been used in packet classification by transforming network traffic data into image-like representations that leverage the spatial hierarchies of the data for the model. A CNN for network traffic classification can achieve high accuracy while reducing false positives far below than other traditional approaches. Similarly, LSTMs perform well on temporal data, enabling the detection of attacks based on temporal features. LSTMs perform very well in terms of identifying each type of attack while addressing the issue of temporal dependencies common in network traffic [11, 12].

D. Hybrid Models and Recent Developments

More recently, to enhance detection capability, research has focused on hybrid models leveraging the combined strength of multiple approaches. This involves further developing deep learning with traditional machine learning methods, anomaly detection, and even rule-based systems to create more robust IDS solutions [13-15]. For example, in [11], a hybrid model combining CNNs and traditional anomaly detection methods was used to achieve better detection rates and lower false positives in complex network environments.

TABLE I. PREVIOUS STUDIES

Study	Approach	Dataset	Accuracy	Issue	Gap
[16]	SVM, RF	KDD Cup 1999	High	Static features, no temporal data	Limited generalization
[17]	LSTM	UNSW-NB15	High	Temporal data captured	Dataset-specific
[18]	CNN + Anomaly	Complex Network Data	Improved	Hybrid approach, reduced false positives	Complex implementation

Despite advances in deep learning for IDS, problems remain. Most approaches operate on single datasets, which makes generalizing their findings extremely narrow. In addition, among the most important problems of deep learning models is the interpretability concern, which is particularly relevant to security applications since, in general, one will face situations where he must understand why something was detected. Although traditional and machine learning-based IDSs laid the foundation for intrusion detection, the integration of deep learning techniques represents a quantum leap in addressing the complexities thrown up by modern cyber threats. This study examines a series of deep learning architectures and their performance on time-series data, with the ultimate goal of improving the accuracy and reliability of IDSs.

III. METHODOLOGY

Figure 1 provides an overview of the proposed system. The architecture is composed of three main components: packet input and preprocessing, followed by deep learning models.

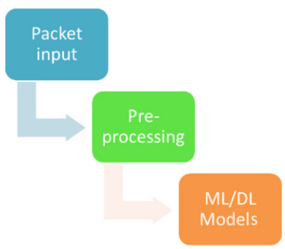

Fig. 1. Key components of a preemptive IDS architecture.

A. Dataset

The UNSW-NB15 dataset was chosen, taken from Kaggle, which consists of both benign and malicious network traffic. It provides a rich source of network traffic records in time series format, capturing diversified network activities. Since this dataset realistically represents modern network traffic, it suits well the evaluation of an IDS, offering a wide variety of attack scenarios to further enhance detection performance.

B. Data Preprocessing

Data preprocessing is very crucial to prepare the dataset for the efficient training of deep learning models. This involved two major steps, namely:

- Data Cleaning: Handle missing or inconsistent values to maintain data integrity, followed by normalization to standardize features across the dataset.
- Data Split: Split that data into three different sets for training, validation, and testing, to offer a balanced evaluation and avoid overfitting.

C. Deep Learning Models

Different deep-learning models were examined to enhance the detection capability of an IDS, each having its strengths in analyzing time series data and network activity patterns:

- Recurrent Neural Networks (RNNs) are considered perfect for sequential data given their temporal dependence on input. They are suitable for capturing temporal patterns within network traffic.
- Long Short-Term Memory (LSTM) networks are a form of RNN that minimizes the vanishing gradient problem. Thus, the model can learn long-term dependencies, finding complicated and continuous patterns of intrusive events.
- Convolutional Neural Networks (CNNs), while typically used on image data, can also be applied to time series data as a multidimensional array. Feature extraction via CNNs is good for identifying spatial hierarchies in network traffic data.

D. Model Training and Evaluation

The model was trained on the preprocessed dataset considering the training and validation data to find an optimal between accuracy and overfitting. Figure 2 shows the flow of this investigation. Each model was evaluated based on precision, recall, and F1-score to measure its effectiveness in picking intrusion patterns against the balance of false positives and false negatives. In addition, a confusion matrix was used, graphically showing the detection results in true positives, true negatives, false positives, and false negatives to provide an overall sense of the effectiveness of the model. F1 score, precision, and recall were used to measure the performance of the IDS model, as they provide an overview of model effectiveness. Precision is the accuracy in predicting items as positive, while recall measures how well a model identifies all instances of interest. The F1 score combines both metrics to provide a single measure of performance, especially useful when dealing with imbalanced datasets. These metrics are used to test the reliability and robustness of IDSs.

E. Hybrid Model

This is an effective model for intrusion detection that amalgamates the convolution and time-series processing layers of machine learning. The architecture of the model initiates the process with two convolution layers acting on feature data, compressing and refining the 44-feature dataset. Additionally, a max-pooling layer emphasizes the key features without changing the time steps. The output from the encoder flows through two LSTM layers, with a self-attention layer in between that selects the most informative historical patterns. This attention layer highlights the important information at each time step to help the model attend to crucial events. Finally, this is passed through a fully connected layer that provides an output as a binary label, indicating the presence of an attack. Figure 3 shows the structure of the multivariate time series intrusion detection model.

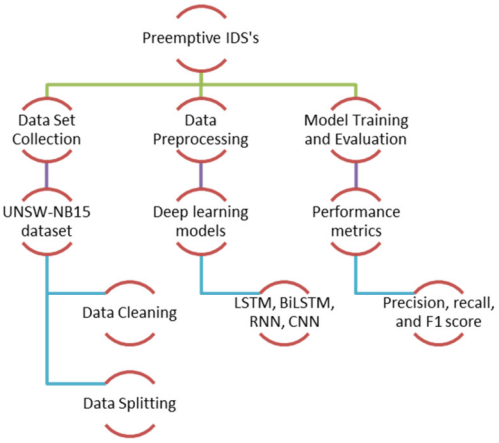

Fig. 2. Research process for preemptive IDS.

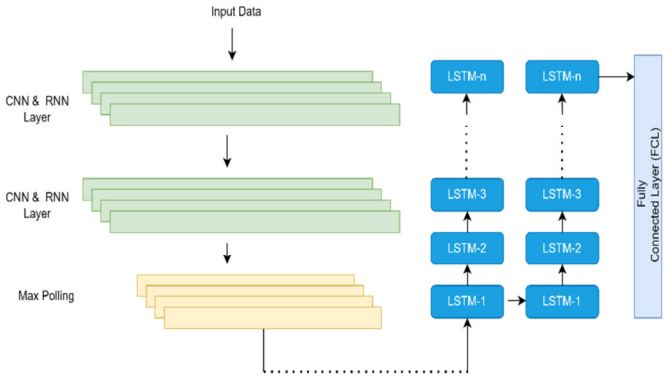

Fig. 3. Structure of the multivariate time series intrusion detection model.

IV. RESULTS AND DISCUSSION

Intrusion detection was examined using different deep learning architectures: LSTM, CNN, RNN, and hybrid combinations. The LSTM model had a 0.79 F1 score, with high precision at 0.88 and moderate recall at 0.70, indicating that its classification abilities were strong but with some misses. The performance of the CNN was slightly lower, with an F1 score of 0.73, a precision of 0.83, and a recall of 0.64, underlining its

problem with false negatives. The RNN also yielded quite similar results, with an F1 score of 0.76, a precision of 0.85, and a recall of 0.69. Hybrid models did much better. CNN+LSTM and CNN+RNN+LSTM had scores of 0.84 and 0.86, respectively, with much better precision of 0.90 and 0.92 and recall values of 0.77 and 0.79. This indicates that the integration of convolutional and recurrent architectures develops their strengths in the quest for more effective intrusion detection systems and thereby amplifies the efficiency of hybrid approaches in improving cybersecurity measures.

TABLE II. EVALUATION RESULTS OF DEEP LEARNING MODELS FOR TIME-SERIES ARCHITECTURES

Model	F1 score	Precision	Recall
LSTM	0.79	0.88	0.70
CNN	0.73	0.83	0.64
RNN	0.76	0.85	0.69
CNN + LSTM	0.84	0.90	0.77
CNN+RNN+LSTM	0.86	0.92	0.79

Figure 4 describes the performance trend for various deep learning models across the three metrics. The x-axis lists several models: LSTM, CNN, RNN, CNN+LSTM, and CNN+RNN+LSTM. On the y-axis, score values go from 0 to 1. In this plot, each metric is represented by a different line. The chart shows that the combination of models is very effective since there is a tendency for the scores to go upward when using combinations such as CNN+LSTM. This visualization makes it quite easy to see how different model architectures affect each metric, providing evidence for the performance gains through model integration.

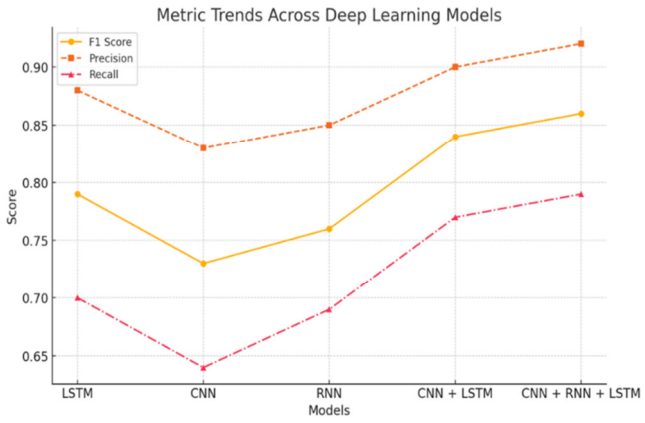

Fig. 4. Performance trends of deep learning models across evaluation metrics.

Figure 5 provides an overall view of the performance of various deep learning models, where the y-axis provides a cumulative score summed across the F1 score, precision, and recall. Each bar is segmented to show the contribution of each metric toward the total performance score. This graph clearly indicates that the proposed hybrid CNN+RNN+LSTM model outperformed its competitors by a remarkably higher cumulative score, highlighting its effectiveness in modeling complex time-series patterns, yielding better predictive accuracy and robustness in all metrics considered.

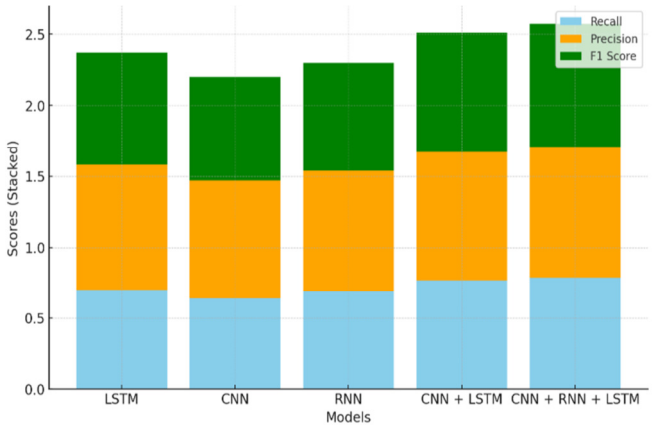

Fig. 5. Performance comparison of deep learning models on key metrics (F1-score, precision, recall).

Figure 6 also shows that hybrid deep-learning models performed better. CNN+RNN+LSTM achieved the highest results for the metrics, with F1, precision, and recall scores of 0.86, 0.92, and 0.79, respectively. This underlines their advantage in intrusion detection, as it combines the complementary strengths of CNN feature extraction and RNN sequence modeling with LSTM's long-term memory retention for a more robust model. The visual format facilitates intuitive comparisons of the strengths and relative weaknesses of the models. For instance, the CNN lighter tone in the recall score (0.64) presents a comparative shortfall, indicating that there is room for optimization. The heat map is a strategic tool to evaluate the performance profile of each model, allowing researchers and practitioners to make informed decisions about the most suitable architecture based on performance.

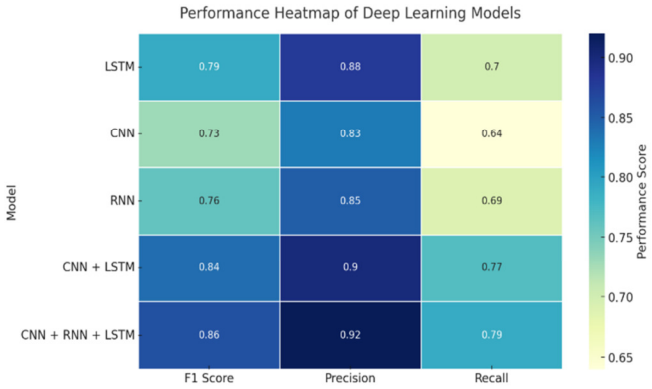

Fig. 6. Deep learning models' performance heatmap.

Figure 7 plots the rate of false positives of the models, comparing single models with hybrid ones. The cells indicate whether the model in a given row has a higher or lower false positive rate than the model in a given column (in red, the lower false positive rate refers to a better performance, while in blue, the higher false rate refers to a worse performance).

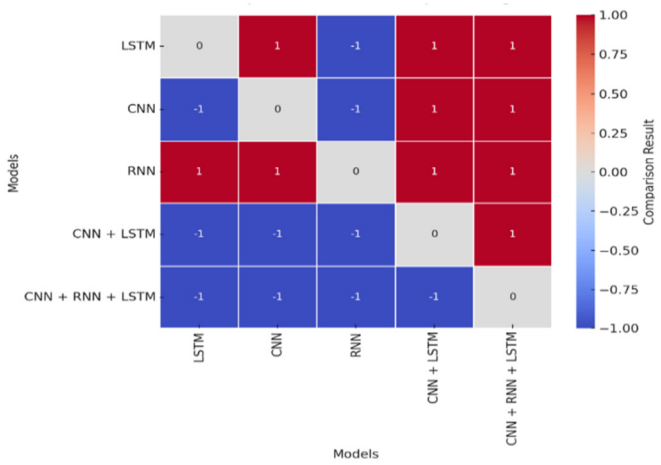

Fig. 7. Comparison matrix.

A. *Analysis of False Positive Rates*

Hybrid models tend to perform better and yield fewer false positives, especially the CNN+RNN+LSTM model, which often appears in red with other cells, indicating that it tends to have a lower false positive rate compared to other models. This corresponds to the higher precision and recall of the hybrid model, which further supports it with fewer positive misclassification cases. Considering the standalone models, the false positive rates for LSTM and RNN are moderately low, but CNN has a relatively higher false positive rate. These facts are reflected in the moderately low number of red cells compared to CNN, resulting in frequent blue cells.

B. *Implications*

This matrix suggests that, in general, hybrid models tend to be better at reducing false positives, probably because complex data patterns can be captured due to the integration of multiple architectures. In this way, hybrid models are more suitable for applications that require high precision and lower misclassification rates.

C. *Computational Costs*

Hybrid deep learning models may require high computing power and memory. The proposed IDS model improved efficiency by using lightweight neural network architectures. It also utilized techniques like batch processing and pruning, which in turn reduce computation overhead. Thus, the model here remains well-suited for real-time applications and balances rich accuracy with practical deployment constraints.

V. CONCLUSION

This study presented a new approach to intrusion detection by combining LSTM, CNN, and RNN architectures, achieving an impressive accuracy of 0.86. The hybrid model leverages the strengths of LSTM to capture temporal dependencies and RNN to handle sequential data, providing significant improvements in detection accuracy and computational efficiency compared to previous methods. This research adds to existing knowledge by proposing a lightweight but effective deep-learning solution tailored for resource-constrained environments, such as IoT devices, where real-time intrusion monitoring is critical. Future work should focus on further optimizing these models, addressing evolving intrusion tactics, and enhancing deployment feasibility in real-world systems.

REFERENCES

[1] H. Liu and B. Lang, "Machine Learning and Deep Learning Methods for Intrusion Detection Systems: A Survey," *Applied Sciences*, vol. 9, no. 20, Jan. 2019, Art. no. 4396, https://doi.org/10.3390/app9204396.

[2] A.-R. Al-Ghuwairi, Y. Sharrab, D. Al-Fraihat, M. AlElaimat, A. Alsarhan, and A. Algarni, "Intrusion detection in cloud computing based on time series anomalies utilizing machine learning," *Journal of Cloud Computing*, vol. 12, no. 1, Aug. 2023, Art. no. 127, https://doi.org/10.1186/s13677-023-00491-x.

[3] M. Sajid *et al.*, "Enhancing intrusion detection: a hybrid machine and deep learning approach," *Journal of Cloud Computing*, vol. 13, no. 1, Jul. 2024, Art. no. 123, https://doi.org/10.1186/s13677-024-00685-x.

[4] R. Mohammad, F. Saeed, A. A. Almazroi, F. S. Alsubaei, and A. A. Almazroi, "Enhancing Intrusion Detection Systems Using a Deep Learning and Data Augmentation Approach," *Systems*, vol. 12, no. 3, Mar. 2024, Art. no. 79, https://doi.org/10.3390/systems12030079.

[5] A. Aldallal, "Toward Efficient Intrusion Detection System Using Hybrid Deep Learning Approach," *Symmetry*, vol. 14, no. 9, Sep. 2022, Art. no. 1916, https://doi.org/10.3390/sym14091916.

[6] P. Wang, X. Song, Z. Deng, H. Xie, and C. Wang, "An Improved Deep Learning Based Intrusion Detection Method," in *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, Chengdu, China, Dec. 2019, pp. 2092–2096, https://doi.org/10.1109/ICCC47050.2019.9064338.

[7] P. Rajesh Kanna and P. Santhi, "Unified Deep Learning approach for Efficient Intrusion Detection System using Integrated Spatial–Temporal Features," *Knowledge-Based Systems*, vol. 226, p. 107132, Aug. 2021, https://doi.org/10.1016/j.knosys.2021.107132.

[8] 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, https://doi.org/10.1109/ACCESS.2019.2895334.

[9] S. Iglesias Pérez, S. Moral-Rubio, and R. Criado, "A new approach to combine multiplex networks and time series attributes: Building intrusion detection systems (IDS) in cybersecurity," *Chaos, Solitons & Fractals*, vol. 150, Sep. 2021, Art. no. 111143, https://doi.org/10.1016/j.chaos.2021.111143.

[10] S. W. Lee *et al.*, "Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review," *Journal of Network and Computer Applications*, vol. 187, Aug. 2021, Art. no. 103111, https://doi.org/10.1016/j.jnca.2021.103111.

[11] Y. C. Wang, Y. C. Houng, H. X. Chen, and S. M. Tseng, "Network Anomaly Intrusion Detection Based on Deep Learning Approach," *Sensors*, vol. 23, no. 4, Jan. 2023, Art. no. 2171, https://doi.org/10.3390/s23042171.

[12] R. Devendiran and A. V. Turukmane, "Dugat-LSTM: Deep learning based network intrusion detection system using chaotic optimization strategy," *Expert Systems with Applications*, vol. 245, p. 123027, Jul. 2024, https://doi.org/10.1016/j.eswa.2023.123027.

[13] T. Su, H. Sun, J. Zhu, S. Wang, and Y. Li, "BAT: Deep Learning Methods on Network Intrusion Detection Using NSL-KDD Dataset," *IEEE Access*, vol. 8, pp. 29575–29585, 2020, https://doi.org/10.1109/ACCESS.2020.2972627.

[14] R. H. Altaie and H. K. Hoomod, "An Intrusion Detection System using a Hybrid Lightweight Deep Learning Algorithm," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 16740–16743, Oct. 2024, https://doi.org/10.48084/etasr.7657.

[15] R. Kaur and N. Gupta, "Harnessing Decision Tree-guided Dynamic Oversampling for Intrusion Detection," *Engineering, Technology & Applied Science Research*, vol. 14, no. 5, pp. 17456–17463, Oct. 2024, https://doi.org/10.48084/etasr.8244.

[16] Z. Wang, D. Jiang, L. Huo, and W. Yang, "An efficient network intrusion detection approach based on deep learning," *Wireless Networks*, Jul. 2021, https://doi.org/10.1007/s11276-021-02698-9.

[17] S. More, M. Idrissi, H. Mahmoud, and A. T. Asyhari, "Enhanced Intrusion Detection Systems Performance with UNSW-NB15 Data Analysis," *Algorithms*, vol. 17, no. 2, Feb. 2024, Art. no. 64, https://doi.org/10.3390/a17020064.

[18] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, 2021, Art. no. e4150, https://doi.org/10.1002/ett.4150.

[19] "UNSW_NB15." Kaggle, [Online]. Available: https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15.