Deep Learning for Intrusion Detection in Reconfigurable Wireless Network

Venkata Naga Rani Bandaru
Assistant Professor
Information Technology
Vishnu Institute of Technology
Bhimavaram, AP
venkatanagarani.b@vishnu.edu.in

Nukaraju Neradabil
Information Technology
Vishnu Institute of Technology
Bhimavaram, AP

neradabillinukaraju@gmail.com

Sangepu Tejaswini
Information Technology
Vishnu Institute of Technology
Bhimavaram, AP

sangeputeja69@gmail.com

Thota Mery Sowmya
Information Technology
Vishnu Institute of Technology
Bhimavaram, AP

merysowmyathota@gmail.com

Peeka kiran Kumar
Information Technology
Vishnu Institute of Technology
Bhimavaram, AP

kirankumarkirky@gmail.com

Abstract : Nowadays Challenges arise in intrusion detection in Reconfigurable Wireless Networks (RWNs), as these networks consist of dynamic topologies, support diverse communication protocols, and involve sophisticated cyber threats or routing attacks. The performance of traditional rule-based systems can even be dubious when it comes to detecting and adequately responding to such evolving attacks. For this reason, this project proposes a strong DL-based intrusion detection system capable of performing both binary classification to decide the presence of an attack and multi-level classification to decide which threat it is, such as DoS attacks, spoofing, unauthorized access, and many others. This DL-IDS uses advanced neural network architectures like CNNs, combined with machine learning models, that enhance the accuracy of the detection and provide a comprehensive security framework. These models are designed to learn the complex attack patterns and dynamic behaviors of the network. Optimized for deployment in resource-constrained environments typical of RWNs, the system not only enhances security but also promptly alerts network administrators to detected threats. This ensures timely precautions and mitigations to protectcommunication systems from cyber threats, maintaining the integrity and reliability of next-gen wireless networks.

Keywords: Cyber Threat Detection, Reconfigurable Wireless Networks (RWNs), Unauthorized Access, Routing attacks

1. INTRODUCTION

With the advanced technologies of wireless communications and continually increased deployment of RWNs, new opportunities have emerged regarding pathways for design flexibility and scalability. Network parameters such as their reconfigurability at topological, frequency levels, and powers provide dynamic scope to modify every aspect of an RWN so as to find itself well-suitable for lots of applications including IoT, Smart cities, autonomous systems, although it brings vast security challenges since its very character of dynamic, heterogeneous networks endows it. With the increased attack surface and evolving cyber threats, incorporating effective intrusion detection mechanisms to safeguard the integrity and reliability of RWNs becomes imperative.The traditional intrusion detection system for the wireless network operates on a rule-based methodology mostly, which can be a rather bad method owing to the high sophistication of attack and zero-day threats. This system is in no way dynamic to the alteration of the manner of attack as well as a change in layout of the network. Deep learning can learn a very complex set of patterns that are deeply hidden in a large number of data. It is possible without explicit human reasoning through self-experience by the machine itself. In the paper, a deep learning-based intrusion detection system has been developed that conducts both binary and multi-level classification to detect attacks. A binary model will classify if there is an attack and the multi-level classification dictates which of these attacks they are; DoS attacks, spoofing, and unauthorized access.

This system, based on using advanced architectures in neural networks with a utilization of CNNs and ML models, allows the proposal of an advanced scheme for wide varieties of cyber threats detection in RWNs. The whole system is optimized for a resource-constrained environment and can even survive critical computationally intensive scenarios. This places the proposed DL-IDS at an advantage of real-time identification and response to emerging possible threats, aspects that further add up to protect next-generation wireless communication systems. The following sections outline the design, implementation, and evaluation of the DL-IDS framework for describing potential applications in the struggle against security issues as presented by dynamic wireless networks.

2. LITERATURE SURVEY

Jabez and Muthukumar [1] discussed an anomaly detection method in Intrusion Detection Systems (IDS) using the Neighborhood Outlier Factor (NOF) approach. This method is designed for distributed environments and focuses on identifying intrusions by detecting outliers in the network traffic. However, their approach faces limitations in detecting known attacks and struggles with real-time scalability, which affects its practical application in dynamic systems.

Abdullah and Abd-alghafar [2] explored the application of Genetic Algorithm (GA) in Intrusion Detection Systems (IDS) to filter traffic data and classify network behaviors as normal or abnormal using a rule-based system. Despite its potential, the system faces challenges in handling dynamic networks, and the simplicity of the rules often fails to detect more complex attack patterns, limiting its effectiveness in real-world scenarios.

Denning [3] presented an Intrusion Detection Model that monitors audit records to detect security violations by identifying abnormal patterns. The model utilizes statistical models and profiles to analyze and flag anomalies. However, it is limited in its ability to detect only abnormal patterns and is dependent on the availability and accuracy of audit records, which may not always reflect the true state of the network.

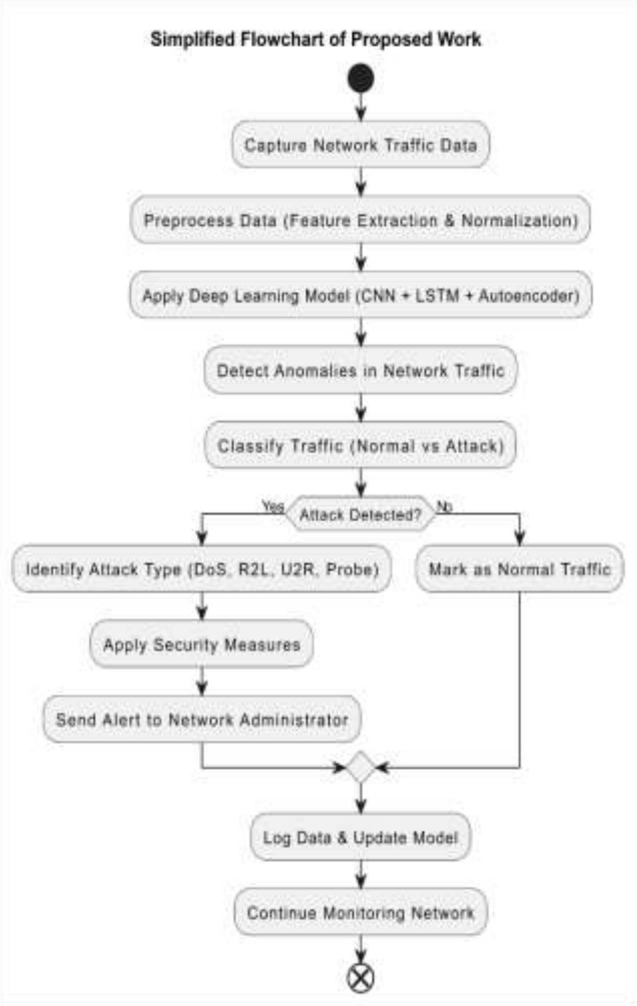
Goyal and Kumar [4] developed GA-NIDS, a Genetic Algorithm-based Network Intrusion Detection System that classifies harmful network connections by generating rule sets based on network features. While effective in some cases, the system struggles with complex network environments, and its reliance on static rule generation limits its ability to adapt to evolving attack vectors, reducing its robustness in real-time applications.

Jaiganesha and Sumathi [5] analyzed intrusion detection using a Back Propagation Neural Network (BPN), focusing on classifying user activities as normal or malicious. Their approach offers a method to predict user behaviors and detect anomalies. However, it faces limitations in multi-level classification and may struggle to capture the full complexity of certain attack patterns, reducing its effectiveness in diverse network environments.

Zhang and He [6] discussed AI-based solutions for intrusion detection in next-generation wireless networks by employing hybrid deep learning frameworks, combining Convolutional Neural Networks (CNN) and autoencoders. Their approach enhances classification accuracy and robustness in detecting complex attack patterns. The use of hybrid deep learning techniques offers significant improvements in detecting intricate intrusion patterns, making it more suitable for modern, dynamic network environments.

3. PROPOSED FRAMEWORK

The proposed Deep Learning-based Intrusion Detection System (DL-IDS) addresses challenges in dynamic Reconfigurable Wireless Networks (RWNs). Based on its framework, real-time traffic benchmark datasets are followed by preprocessing steps on normalization and selection of features so that the models can learn perfectly. The techniques used are such deep learning algorithms applied for binary, multi-level classifications, like finding DoS threats, spoofing threats, and unauthorized accesses, with applications of CNN, RNN techniques. Optimization techniques include dropout and hyperparameter tuning, ensuring that the model performs efficiently within resource-constrained environments. A real-time detection and alert mechanism can notify administrators promptly and initiate automatic responses to the threats. Continuous learning ensures that it adapts to new threats while performance evaluation in terms of accuracy and recall ensures system effectiveness for the next generation of wireless.


Simplified Flowchart of Proposed Work
Capture Network Traffic Data
Preprocess Data (Feature Extraction & Normalization)
Apply Deep Learning Model (CNN + LSTM + Autoencoder)
Detect Anomalies in Network Traffic
Classify Traffic (Normal vs Attack)
Attack Detected?
Yes
No
Identify Attack Type (DoS, R2L, U2R, Probe)
Mark as Normal Traffic
Apply Security Measures
Send Alert to Network Administrator
Log Data & Update Model
Continue Monitoring Network

4. METHODOLOGY

First, the methodology acquires the network traffic data from both normal scenarios and those involving attacks. The data included carries various characteristics such as connection duration, protocol type, byte counts, and many others. The whole process is about pre-processing, which aims at cleaning and preparing the data for analysis purposes. This would include normalization whereby it scales feature values into a specific range in order to standardize and present uniformity while using feature selection about minimizing all the useless data yet ensure critical information remains.

Based on these capabilities, the deep learning model relies on CNNs integrated with some other machine learning techniques like Linear Support Vector Classifier, Logistic Regression, Gaussian Naïve Bayes, and Bootstrap Aggregation known as Bagging. Both are used for detection and classification of attacks. It uses the capability of CNNs to extract meaningful patterns from the data; it exploits the capability of the models for the detection of spatial relationships as well as hierarchical relationships within features. The system, as a whole, performs two classification tasks. First, binary classification classifies the network connection as normal or malicious. Then, multi-level classification identifies which type of attack is involved, whether it is DoS, R2L, Probe, or U2R. This multi-level classification improves the intrusion detection granularity, providing more accurate responses to potential threats.

Optimization techniques are brought to bear such that the aptness of this model for environment resource-constraint typical of the RWN would be ensured through dropout regularization, through which overfitting is lowered by temporarily disabling any random parts in the network and hyperparameter tunings, used to fine-tune model parameters such as learning rate size of the networks, and even the number epochs a model might have seen over time, into delivering better performance.

The system deployed gives real-time intrusion detection and alerts. It continuously monitors network traffic, analyzing data in respect of irregular patterns to trigger alerts with suspicious activity observed. This means the network administrators can take immediate action, reducing the chance that such attacks can affect networks. This system can be effective in detection as well as classification of various types of intrusions through advanced deep learning methods. In other words, this can be a robust solution toward maintaining security in the next generation of wireless networks.

4.1 Dataset Description

4.1.1 Features of the NSL-KDD Dataset

The dataset consists of 41 features that define the variations in different network connection attributes. These features come under the following categories:

1. Basic Features : Some of the primary attributes are connection length or duration, protocol-type (such as TCP, UDP, ICMP), and service (application-layer protocols such as HTTP, FTP), providing an overall idea of network connection. Features like src_bytes and dst_bytes reflect the volume of data exchanged between source and destination.

2. Content-Based Features: These features center on the content of the connection, which it uses to detect unauthorized access attempts. For instance, num_failed_logins tracks failed login attempts, and logged_in indicates whether a login attempt was successful.

3. Traffic and Temporal Features: Count (connections to the same host) and srv_count (connections to the same service) are features that help identify unusual traffic patterns, such as those seen in Distributed Denial of Service (DDoS) attacks. Flag serves as a status indicator of the connection, helping detect anomalies.

4.1.2 ATTACK CATEGORIES IN NSL-KDD DATASET

The dataset consists of normal connections and four categories of attacks-differences with respect to the different methods of intrusion being used against a system.

1. Denial of Service :

through some means or other, interfere with denial of vital services for crackers by flooding the various available resources in their systems with bandwidth and processing power, for example-dos SYN Flood, dos UDP Flood, when very large numbers of requests being generated intended to consume the capacity of the system. count and srv_count are some of the important features for identifying these types of attacks that usually create a high-frequency connection attempt.

2. Remote to local (R2L)

R2L attacks are made by unauthorized access from a remote machine to the local system. These are the attacks that usually take advantage of weak authentication. Password guessing attacks and phishing attempts are some of the representations for R2L attacks. Features like number_failed_logins and logged_in are pretty important in detecting certain patterns tied with successful breaks and repeated login attempts.

3. User to root (U2R)

U2R attacks look so named because the attacker has gained user level access, followed by privilege escalation to attain administrative control over the foreign system (buffer overflow exploitation being classic examples of U2Rs). For such attacks, key features are concerned with the number of conditions that keep track of compromise (i.e., num_compromised) and root_shell, which denotes privilege escalation.

4. Probe

A probe is an act of scanning or monitoring of the system or network for possible attacks. Typically port scanning or network mapping, the attacker uses the probe to ascertain what ports are open and which devices are up. A very high value of count and srv_count, coupled with abnormal patterns of flags, indicates probe behavior.

4.1.3 APPLICATIONS OF NSL-KDD DATASET

The NSL-KDD dataset is extensively used for both binary classification (normal vs. attack) and multi-class classification (specific attack type). Its detailed feature set enables fine-grained detection of network intrusions, supporting hierarchical classification approaches. In binary classification, the goal is to detect whether a connection is normal or malicious. In multi-class classification, the dataset enables models to differentiate between attack categories such as DoS, R2L, U2R, and Probe. This two-step classification approach improves the accuracy and granularity of intrusion detection systems.

4.2 Linear Support Vector Classifier (Linear SVC)

Linear SVC is a machine learning algorithm used for supervised classification. It works by finding the optimal hyperplane that separates data points into different classes with the maximum margin. The mathematical foundation of Linear SVC is based on the hyperplane equation $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b,$ where w is the weight vector,x is the input feature vector, and b is the bias term. The algorithm minimizes the hinge loss function:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \max(0, 1 - y_i(\mathbf{w}^T\mathbf{x}_i + b)),$$

where CCC is the regularization parameter and yiy_iyi are the class labels (±1\pm 1±1).In the context of intrusion detection, Linear SVC is useful for binary classification tasks, such as distinguishing normal network traffic from anomalous activity. Its simplicity and efficiency make it suitable for detecting linearly separable attacks, ensuring low computational overhead in wireless network environments.

4.1.4 Convolutional Neural Networks (CNNs)

CNNs are deep learning architectures designed to process grid-like data, such as images or structured time-series data. The key operation in CNNs is the convolution, defined as:

$$S(i,j) = (I * K)(i,j) = \sum_{m}\sum_{n} I(i+m, j+n)K(m,n),$$

where III is the input matrix (e.g., network traffic data), KKK is the convolution kernel, and S(i,j)S(i, j)S(i,j) is the resulting feature map. CNNs also include pooling layers to reduce spatial dimensions and fully connected layers for classification.

For this project, CNNs are used to analyze structured network traffic data, extracting spatial patterns indicative of specific attack types. By processing features such as packet sizes or connection flags, CNNs can identify subtle anomalies, making them effective for detecting complex intrusions like Distributed Denial of Service (DDoS) attacks.

4.1.5 Long Short-Term Memory Networks (LSTMs)

LSTMs are a specialized type of Recurrent Neural Network (RNN) designed to capture long-term dependencies in sequential data. They rely on gates to regulate the flow of information, including:

1.Forget Gate: which decides what information to discard from the previous state.

2. Input Gate :which determines what new information to store.

$$i_t = \sigma(\mathbf{W}_i \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i)$$

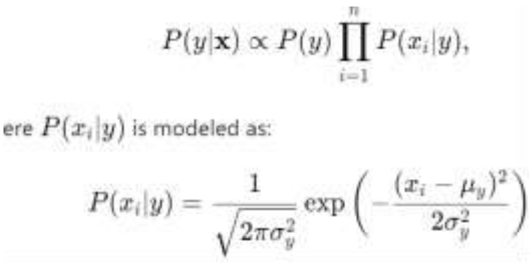
3. Output Gate :

$$o_t = \sigma(\mathbf{W}_o \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o).$$

LSTMs are particularly effective for analyzing time-series data, such as sequential network traffic records.

4.1.6 Gaussian Naïve Bayes

Gaussian Naïve Bayes is a probabilistic classifier based on Bayes' theorem. It assumes that features are conditionally independent and follow a Gaussian distribution. The probability of a class y given a feature vector x is:



$$P(y|\mathbf{x}) \propto P(y) \prod_{i=1}^{n} P(x_i|y),$$

ere $P(x_i|y)$ is modeled as:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Gaussian Naïve Bayes is used for lightweight classification in this project, particularly for scenarios where features (e.g., packet counts or connection durations) are approximately independent. Its probabilistic nature provides a clear confidence measure for intrusion detection.

4.1.6 Bootstrap Aggregation (Bagging)

Bagging is an ensemble learning technique to enhance the performance of the models via a combination of predictions made by various base models trained on bootstrapped subsets of the data.

The mathematical form of ensemble prediction for case of classification is as follows:

where hk(x) is the prediction of the k-th base model. In the present project, Bagging would reduce the variance of individual models like Decision Trees and improve generalization against overfitting. Thus, it aggregates predictions to provide a consistent performance of intrusion detection against different attacks.

$$f_t = \sigma(\mathbf{W}_f \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f),$$

scenarios.

4.1.7 Autoencoders for Intrusion Detection in Wireless Networks

In the context of intrusion detection for secure reconfigurable wireless networks, autoencoders are trained on normal network traffic to learn its latent representation. When presented with anomalous data (e.g., traffic generated by an intrusion), the autoencoder struggles to reconstruct it, resulting in a high reconstruction error. This discrepancy can be used to identify anomalies.

Components :

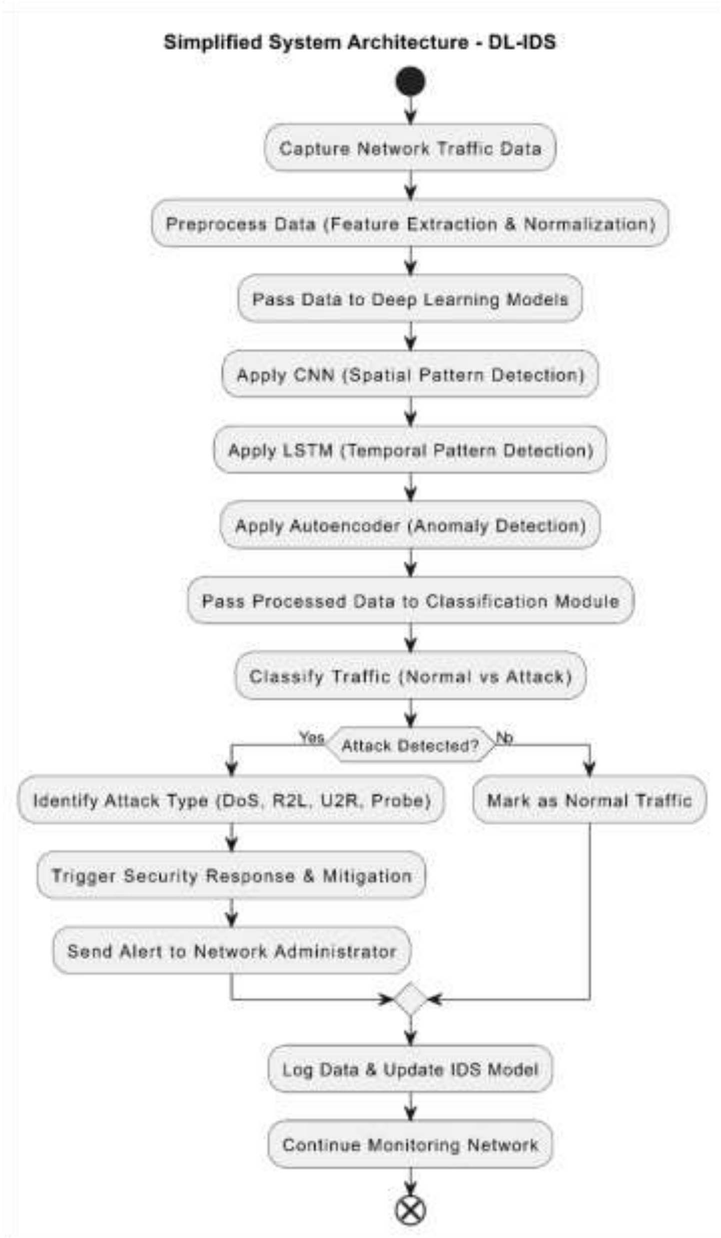
1 . Encoder: Compresses the input data into a latent representation.

$$\mathbf{z} = f_{\text{encoder}}(\mathbf{x}) = \sigma(\mathbf{W}_e\mathbf{x} + \mathbf{b}_e)$$

2. Decoder: Reconstructs the data from the latent representation.

$$\hat{\mathbf{x}} = f_{\text{decoder}}(\mathbf{z}) = \sigma(\mathbf{W}_d\mathbf{z} + \mathbf{b}_d)$$

4.2Architecture


Simplified System Architecture - DL-IDS
Capture Network Traffic Data
Preprocess Data (Feature Extraction & Normalization)
Pass Data to Deep Learning Models
Apply CNN (Spatial Pattern Detection)
Apply LSTM (Temporal Pattern Detection)
Apply Autoencoder (Anomaly Detection)
Pass Processed Data to Classification Module
Classify Traffic (Normal vs Attack)
Yes
Attack Detected?
No
Identify Attack Type (DoS, R2L, U2R, Probe)
Mark as Normal Traffic
Trigger Security Response & Mitigation
Send Alert to Network Administrator
Log Data & Update IDS Model
Continue Monitoring Network

The 41 features include basic attributes, such as connection duration, protocol type, and service type, and derived features that give higher-level insight, such as the number of failed login attempts, connections to the same host, or connections to the same service. Features like num_compromised, root_shell, and is_guest_login help determine unauthorized access attempts and privilege escalation. These are critical in trying to differentiate normal behavior from malicious activity in network traffic.

The dataset can support both the binary classification-a connection is normal or malicious -and multi-class classification-actually identifies the real attack type. Therefore, the dataset is very ideal for training advance machine learning as well as deep learning models with fine-grained intrusion detection.

5. RESULTS

1. Decision Tree Classifier and Random Forest Classifier are remarkably good for binary classification problems with regard to accuracy, precision, recall, and F1 score. Also the accuracy of the model is almost perfect relative to the accuracy of the train dataset; accuracy and F1 scores relative to test dataset varied between 99.83% and 99.85%. Such high-performance performance makes it prudent that such models should be applied where accuracy and reliability are crucial. In both cases, the classifier will outperform the tree simply because it is an ensemble model of great merit. Overfitting, a major problem majorly associated with single decision trees is reduced since a forest builds numerous decision trees then aggregates their predictions for more robustness and generalization capabilities.

1. One major ensemble method that the Random Forest employs is known as bootstrap aggregation, or bagging. It works by training a multitude of decision trees using different subsets of the data set and then produces a final prediction from a weighted average of their output. This procedure minimizes overfitting and promotes model stability in the presence of noisy data. The inherent randomness involved in feature selection during tree building allows Random Forest to capture multiple decision boundaries; this further propels performance in unseen data.

2. On the other hand, Logistic Regression and Gaussian Naïve Bayes are relatively simple models that have achieved only moderate performance, with much lower accuracy and F1 scores than complex models. For example, Logistic Regression relies on the linearity between the features and the target, thereby making it incapable of handling the intricacies in the sense of the non-linear decision boundaries most existing in the network intrusion detection tasks. Similarly, GaussianNB that assumes feature independence and is based on a Gaussian distribution fails to catch intricate patterns and correlations in the data, and thus may suffer from a rather lower predictive accuracy.

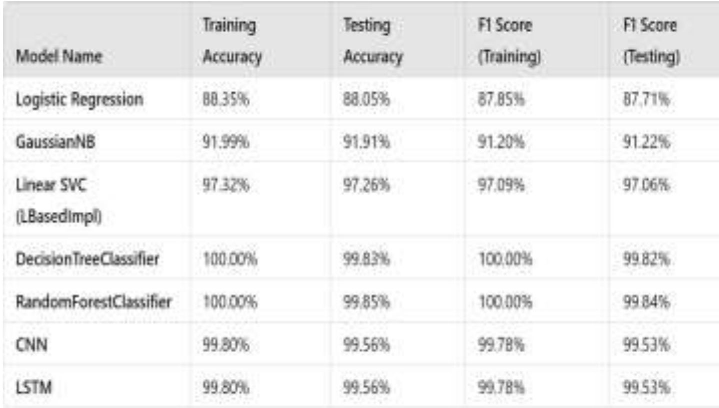
3. However, more complex models such as Linear Support Vector Classifier, Convolutional Neural Networks, and Long Short-Term Memory networks showed excellent performance because accuracy was between 99.5% and 99.8% for both training and testing sets, mainly because of their ability to detect complex relationships in data:

1. Linear SVC: This model can perform well in separating the data points through a hyperplane. The maximization of the margin for classes gives the best decision boundary. Linear SVC performs quite well on linearly separable data points but must be applied several times, along with kernel transformations, in order to solve more complex, nonlinear problems.

2. Convolutional Neural Networks: They are very efficient at finding spatial and hierarchical patterns in the data, making them ideal for feature extraction on this structured dataset. Their convolutional filters can apply these convolutions to learn local patterns while dimensionality is reduced by the pooling layers.

3. Another form of a recurrent neural network is the Long Short-Term Memory networks, or LSTMs. LSTMs have great powers in discovering dependencies within sequential data. They use memory cells that allow them to remember information relevant over long sequences and are, therefore, very well poised for analyzing time-series features within network traffic.

Table 2. Model Accuracy



Model Name	Training Accuracy	Testing Accuracy	F1 Score (Training)	F1 Score (Testing)
Logistic Regression	88.35%	88.05%	87.85%	87.71%
GaussianNB	91.99%	91.91%	91.20%	91.22%
Linear SVC (LBasedImpl)	97.32%	97.26%	97.09%	97.06%
DecisionTreeClassifier	100.00%	99.83%	100.00%	99.82%
RandomForestClassifier	100.00%	99.85%	100.00%	99.84%
CNN	99.80%	99.56%	99.78%	99.53%
LSTM	99.80%	99.56%	99.78%	99.53%

These results imply that Random Forest and Decision Tree are the most accurate models in terms of classification of binary class problems, but CNN and LSTM are also quite strong contenders. In terms of performance, overall Random Forest tends to be the best one both for training and testing because it gives robust performance in all metrics.

The multiple attack detection system classifies intrusions into DoS, R2L, U2R, and Probe attacks using features like connection duration, protocol type, byte counts, and error rates from the NSL-KDD dataset. It even uses 41 network attributes in order to distinguish attack patterns. With this model, it attains 73% classification accuracy efficiently identifying diverse types of attacks for stronger network security.

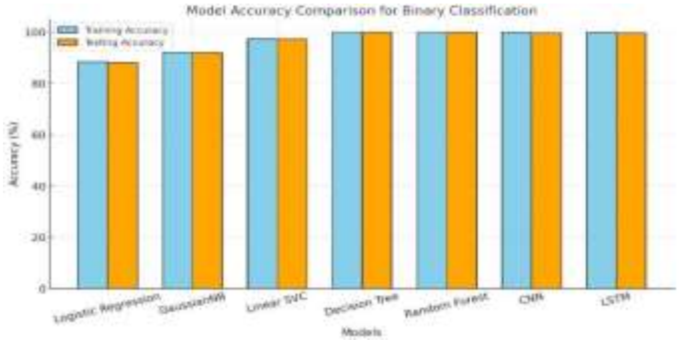


Fig 2: Model Accuracy

6. CONCLUSION

This DL-IDS has been proposed, which is designed to secure the RWNs by using the advanced models like CNNs and RNNs. It uses the NSL-KDD dataset and real-time traffic data for high accuracy intrusion detection, including DoS attacks and unauthorized access.

The system, optimized for dynamic environments and resource-constrained devices, is quite suitable for IoT and smart cities applications. The real-time monitoring of this DL-IDS supports swift threat mitigation. Binary classification is very well performed; multi-level classification might be improved through advanced techniques and better datasets.

This DL-IDS presents an applicable, scalable solution to increase network security for next-generation wireless systems.

7 FUTURE ADVANCEMENTS

The DL-IDS system proposed herein would provide a robust basis for the security of RWNs. However, for the overcoming of new emerging challenges and enhancement in efficiency as well as adaptability, some improvements can be further considered below:

7.1.1 Multi-Level Classification Accuracy Improvement

Applying even more advanced architectures, such as transformers or hybrid deep learning models like CNN-LSTM or CNN-RNN to improve the capability of the system in classifying certain types of attacks. It is using ensemble techniques that combine the strengths of diverse models to better predict outcomes.

8. REFERENCES

[1] S. Widup, A. Pinto, D. Hylender, G. Bassett, and P. Langlois. Data Breach Investigations Report. [Online]. Available: https://www.wired.com/images_blogs/threatlevel/2011/04/Verizon -2011-DBIR_04-13-11.pdf

[2] J. Andrew, R. J. Eunice, and J. Karthikeyan, “An anonymizationbased privacy-preserving data collection protocol for digital health data,” Front Public Health, vol. 11, Mar. 2023.

[3] O. Aslan and R. Samet, “A comprehensive review on malware detection approaches,” IEEE Access, vol. 8, pp. 6249–6271, 2020.

[4] Z. Yang et al., “A systematic literature review of methods and datasets for anomaly-based network intrusion detection,” Comput. Secur., vol. 116, 102675, May 2022.

[5] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, “Survey of intrusion detection systems: Techniques, datasets and challenges,” Cybersecurity, vol. 2, no. 1, 20, Dec. 2019.

[6] M. Masdari and H. Khezri, “A survey and taxonomy of the fuzzy signature-based intrusion detection systems,” Appl. Soft. Comput., vol. 92, 106301, Jul. 2020.

[7] S. Dwivedi, M. Vardhan, S. Tripathi, and A. K. Shukla, “Implementation of adaptive scheme in evolutionary technique foranomaly-based intrusion detection,” Evol. Intell., vol. 13, no. 1, pp. 103–117, Mar. 2020.

[8] 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, Jan. 2021.

[9] D. E. Kim and M. Gofman, "Comparison of shallow and deep neural networks for network intrusion detection," in Proc. 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), 2018, pp. 204–208.

[10] Y. Zhong et al., "HELAD: A novel network anomaly detection model based on heterogeneous ensemble learning," Computer Networks, vol. 169, 107049, Mar. 2020.

[11] Wang, Z., Jiang, D., Huo, L., Yang, W. (2021). An Efficient Network Intrusion Detection Approach Based on Deep Learning. doi: 10.1007/s11276-021-02698-9

[12] Nguyen Dang, K. D., Fazio, P., Voznak, M. (2024). A Novel Deep Learning Framework for Intrusion Detection Systems in Wireless Networks. doi: 10.3390/fi16080264

[13] Singh, A., Amutha, J., Nagar, J., Sharma, S. (2022). A Deep Learning Approach to Predict the Number of k-Barriers for Intrusion Detection Over a Circular Region Using Wireless Sensor Networks. doi: 10.48550/arXiv.2208.11887

[14] Sultan, M. T., El Sayed, H., Khan, M. A. (2023). An Intrusion Detection Mechanism for MANETs Based on Deep Learning Artificial Neural Networks (ANNs). doi: 10.48550/arXiv.2303.08248

[15] Yang, K., Ren, J., Zhu, Y., Zhang, W. (2018). Active Learning for Wireless IoT Intrusion Detection. doi: 10.48550/arXiv.1808.01412

[16] Gupta, A., Pandey, O. J., Shukla, M., Dadhich, A., Mathur, S., Ingle, A. (2021). Computational Intelligence-Based Intrusion Detection Systems for Wireless Communication. doi: 10.48550/arXiv.2105.03204

[17] Zhang, J., He, T. (2022). AI-Based Solutions for Intrusion Detection in Next-Generation Wireless Networks. doi: 10.1109/ACCESS.2022.3140198

[18] Kim, D. E., Gofman, M. (2018). Comparison of Shallow and Deep Neural Networks for Network Intrusion Detection. doi: 10.1109/CCWC.2018.8301638

[19] Zhong, Y., Wang, Y., Wang, L., Li, J., Li, Z. (2020). HELAD: A Novel Network Anomaly Detection Model Based on Heterogeneous Ensemble Learning. doi: 10.1016/j.comnet.2020.107049

[20] Ahmad, Z., Khan, A. S., Shiang, C. W., Abdullah, J., Ahmad, F. (2021). Network Intrusion Detection System: A Systematic Study of Machine Learning and Deep Learning Approaches. doi: 10.1002/ett.4147

[21] S., Riyaz., G., Sannasi. (2020). A Deep Learning Approach for Effective Intrusion Detection in Wireless Networks Using CNN. doi: 10.1007/s00500-020-05017-0

[22] S., Wang., et al. (2019). Intrusion Detection for WiFi Network: A Deep Learning Approach. doi: 10.1007/978-3-030-06158-6_10

[23] Enhanced Intrusion Detection in Wireless Sensor Networks Using Deep Learning. (2024). doi: 10.1007/s11042-024-19305-6

[24] Real-Time Intrusion Detection in Wireless Network: A Deep Learning Approach. (2020). doi: 10.1109/access.2020.9178792

[25] Dinh Nguyen, Khoa., Fazio, P., Voznak, M. (2024). A Novel Deep Learning Framework for Intrusion Detection Systems in Wireless Networks. doi: 10.3390/1999-5903

[26] Kimanzi, R., Kimanga, P., Cherori, D., Gikunda, P. K. (2024). Deep Learning Algorithms Used in Intrusion Detection Systems: A Review. doi: 10.4856/2402-17020

[27] Feasibility of Deep Learning Sensor Network Intrusion Detection. (2019). doi: 10.1109/8653348

[28] Patel, H., Mehta, P., Bhatt, P. (2023). Intrusion Detection Using Federated Deep Learning in IoT Networks. doi: 10.1007/s00521-023-08357-7

[29] Liu, B., Xu, Y., Chen, M. (2021). Intrusion Detection in 5G Networks: A Deep Learning-Based Approach. doi: 10.1109/ACCESS.2021.3085481

[30] Choi, Y., Han, J., Seo, S. (2022). A Lightweight Intrusion Detection Model Using Edge AI for Wireless Sensor Networks. doi: 10.1109/TNSM.2022.315