iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79337

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Integrating CNN & Bidirectional LSTM for Efficient Anomaly Detection In Networks

Dr. R. Obulakonda Reddy[1], M. Varun[2], B. Vigneshwar[3], P. V. Rishyanth[4]
[1]*Professor, Department of Computer Science and Engineering (Cyber Security), Institute of Aeronautical Engineering, Dundigal, Hyderabad - 500043, Telangana, India*
[2, 3, 4]*Department of Computer Science and Engineering (Cyber Security), Institute of Aeronautical Engineering, Dundigal, Hyderabad - 500043, Telangana, India*

Abstract: ***The spread of internet and computer networks has revolutionized information exchange, but it has also introduced significant security risks. Cyber attackers exploit vulnerabilities within these networks to compromise data confidentiality, integrity, and availability. Detecting anomalous activities in networks is critical for maintaining security but presents challenges due to the volume of data and the complexity of attacks. The objective of this study is to develop an effective anomaly detection system for computer networks using deep learning techniques. Specifically, we aim to design a Convolutional Neural Network with Bidirectional Long Short-Term Memory (CNN Bi-LSTM) model. The goal is to achieve high accuracy in identifying network-based anomalies while considering various hyperparameters and optimizing model performance. The approach involves designing and training a CNN Bi-LSTM model for network anomaly detection. We experiment with different hyperparameters, including optimizers (Nadam, Adam, RMSprop, Adamax, SGD, Adagrad, Ftrl), epochs, batch size, and learning rate. We utilize the NSL-KDD and UNSW-NB15 datasets for training and evaluation. Performance metrics such as accuracy and F1-score are used to assess the effectiveness of the model. The CNN Bi-LSTM model demonstrates outstanding performance in detecting network anomalies, achieving high accuracy . Through meticulous analysis of hyperparameters, we identify the optimal configuration that maximizes detection accuracy. Comparative analysis with existing anomaly detection methods confirms the superiority of our proposed approach. In conclusion, our study highlights the efficacy of deep learning, particularly CNN Bi-LSTM models, in detecting network-based anomalies. And also added CNN and a hybrid CNN-LSTM method are implemented to improve prediction accuracy, with the CNN-LSTM achieving an impressive 99% accuracy rate. A user-friendly front end using Flask is developed, allowing easy access for testing, with integrated user authentication for secure access, enhancing usability and reliability.***

Index terms: ***Network Intrusion Detection System, Machine Learning, Deep Learning,*** **CNN Bi-LSTM,** ***NSL-KDD.***

I. INTRODUCTION

With the advent of information and technology, the way we transmit and handle information has shifted dramatically. Nowadays, essential data—including audio, visual content, and various digital records such as banking or personal information—travels in the form of binary data from one point to another. Alongside this progression, protecting data from unauthorized access has become a critical concern, leading to the development of tools aimed at detecting and blocking intrusions. One vital concept in this context is the idea of an anomaly—a deviation from the expected pattern in a dataset. These unusual data points, sometimes called outliers, often signal potential issues or threats. Because they don't conform to typical behavior, specific detection methods must be applied to identify them. The ability to spot these anomalies is essential across a range of industries, including fraud prevention, system diagnostics, cybersecurity, healthcare, and business intelligence. In the realm of cybersecurity, anomaly detection plays a pivotal role. It helps defend against unauthorized access or data breaches by identifying activities that violate core principles of digital security: Confidentiality, Integrity, and Availability—collectively known as the CIA triad. A system is only considered secure if these three pillars are properly upheld. Intrusion Detection Systems (IDS) serve as the backbone of digital surveillance, analyzing network and system behavior to flag potential threats. Broadly, these systems fall into two categories: Signature-Based IDS (SIDS) and Anomaly-Based IDS (AIDS). While SIDS rely on known patterns of malicious activity (signatures), AIDS detect unknown or novel threats by learning the difference between normal and abnormal behavior. AIDS techniques can be further categorized based on their source of data: host-based, which monitor activity within individual devices, and network-based, which scan entire networks for suspicious behavior. In terms of methodology, anomaly detection may follow supervised, unsupervised, or semi-supervised approaches—depending on whether the system is trained on labeled data, unlabeled data, or a mix of both.

Unlike signature-based systems, anomaly-based detection models—especially those powered by machine learning (ML) or deep learning (DL)—can adapt and identify threats without relying solely on predefined signatures. In particular, DL techniques offer significant advantages in learning complex patterns within large datasets, making them especially valuable for today's expansive digital environments.

Traditional ML models, while useful and relatively simple to implement, often perform best with smaller, well-prepared datasets. Their success hinges on effective feature engineering and data splitting strategies. However, when applied to high-dimensional data like images or natural language, these conventional methods may struggle to deliver reliable results.

This is where deep learning architectures such as Convolutional Neural Networks (CNNs) come into play. CNNs are adept at processing image data, where initial network layers detect low-level features (such as lines and textures), which are then combined into more complex representations in deeper layers—eventually leading to accurate classification or prediction.

In contrast, Long Short-Term Memory (LSTM) networks excel at handling sequential data. These models are particularly useful for tasks involving time series or textual data, as they can preserve long-term dependencies in the data. A variant known as Bidirectional LSTM (BiLSTM) improves this capability by analyzing sequences in both forward and backward directions, reducing the risk of forgetting important contextual information.

DL models also scale better with large datasets and high-dimensional features than conventional ML algorithms. However, their performance depends on several key factors, including the number of hidden layers, the type and number of neurons, the choice of activation functions, the batch size, and the number of training iterations (epochs). When designed carefully, these models can significantly outperform traditional systems in identifying anomalies and enhancing overall system security.

II. LITERATURE SURVEY

This work explores the evolution of the CIA triad—Confidentiality, Integrity, and Availability—by examining both practitioner and scholarly perspectives. Professionals in information security have long embraced the triad as a foundational model for protecting data systems. However, academics have raised concerns about its limitations, particularly its inability to fully address emerging socio-technical complexities in modern security environments. By revisiting the fundamental principles of the CIA framework, this paper attempts to bridge the gap between these two views. The central argument suggests that while the triad remains highly relevant, its continued use is not due to a rejection of more holistic or human-centered enhancements, but rather its ability to evolve and integrate broader concepts under a more expansive interpretation. The study concludes by outlining potential avenues for further academic inquiry into this reconceptualized framework.

As cyber threats continue to grow in frequency and complexity, Network Anomaly Detection Systems (NADS) have become essential components in cybersecurity infrastructures. These systems are designed to detect and mitigate unusual behaviors that may indicate a breach. The paper investigates multiple facets of anomaly-based Network Intrusion Detection Systems (NIDS), highlighting modern cyberattack strategies and how they align with the cyber kill chain model. It also evaluates the effectiveness of various Decision Engine (DE) mechanisms, including newer approaches that incorporate ensemble and deep learning methods. To support these techniques, the study references prominent benchmark datasets commonly used to train and test such models. Applications of these systems extend to critical areas such as data centers, the Internet of Things (IoT), and cloud-based services including fog computing. A set of experimental insights is shared, accompanied by recommendations for promising future research directions.

Machine learning (ML) has been widely adopted in intrusion detection but often suffers from challenges such as low detection accuracy and reliance on extensive manual feature engineering. To overcome these issues, a new model called Deep Learning Network Intrusion Detection (DLNID) is proposed. This architecture combines an attention mechanism with a Bidirectional Long Short-Term Memory (Bi-LSTM) network. The process begins with feature extraction using a Convolutional Neural Network (CNN), which captures key spatial features from traffic data. The attention mechanism then emphasizes more relevant features across the channels, and the Bi-LSTM component processes temporal patterns in the sequence. Recognizing the imbalance in public datasets for intrusion detection, the model integrates Adaptive Synthetic Sampling (ADASYN) to augment underrepresented classes and applies a customized stacked autoencoder to reduce dimensionality. When tested on the NSL-KDD benchmark dataset, the model achieved an accuracy of 90.73% and an F1 score of 89.65%, outperforming several existing approaches.

Intrusion Detection Systems (IDS) remain vital in identifying and blocking unauthorized network activities. However, the dynamic nature of network environments often results in malicious traffic being overshadowed by normal behavior, creating class imbalance problems that hinder model training and increase false positives. To address this, another approach incorporates a hybrid sampling technique with a deep hierarchical network architecture.

First, One-Side Selection (OSS) is used to filter noise from the dominant class. Then, Synthetic Minority Over-sampling Technique (SMOTE) balances the dataset by enriching the minority class. This helps the model better recognize attack patterns. The architecture leverages CNNs to extract spatial characteristics and BiLSTM layers to learn temporal dependencies. The model's effectiveness was demonstrated using the NSL-KDD and UNSW-NB15 datasets, achieving classification accuracies of 83.58% and 77.16%, respectively.

In recent years, machine learning has become a popular tool for intrusion detection due to its ability to uncover unknown or evolving threats. However, one major limitation is the scarcity of labeled attack data compared to the abundance of normal traffic data. This imbalance can severely limit the effectiveness of ML models. To tackle this, a novel strategy involving Generative Adversarial Networks (GANs) is introduced. GANs are used to generate synthetic attack data, which is then merged with the original dataset to create a more balanced training set. Three common ML algorithms were trained using this augmented dataset, and tests across several standard IDS datasets—as well as a custom dataset—showed significant improvements in detection performance. Furthermore, visualization techniques were applied to analyze the characteristics of the generated samples and their impact on model behavior, confirming the potential of GANs in improving IDS accuracy.

III. METHODOLOGY

A. Proposed Work

The system introduces a CNN Bi-LSTM model in network-based anomaly detection, comparing different hyperparameters including optimizers, epochs, batch size, learning rate, and network architecture to surpass the current approaches. Experiments on NSL-KDD and UNSW-NB15 datasets test the efficacy of the model to detect anomalies. And also CNN, and a combination technique utilizing Convolutional Neural Network (CNN) [13] and Long Short-Term Memory (LSTM) are used to improve prediction accuracy, and CNN with LSTM has a high accuracy of 95%. Moreover, a simple front end is created utilizing the Flask framework to facilitate access for user testing. In addition, user authentication facilities are incorporated to make the system available securely, making it more usable and reliable for applications in actual scenarios.

B. System Architecture

The overall proposed model encompasses the following steps.

Step-1 Data Collection and Modelling

Step-2 Data Pre-procession

Step-3 Prepare the training and testing dataset

Step-4 Train and Test the Bi-LSTM [14, 15] Model

Step-5 Model Evaluation and anomaly detection

Step-6 Model Compare and Decision

The overall implementation schematic of the Bidirectional LSTM-based model is given in Fig. 1. A detailed discussion of the above-stated methods is provided in the subsequent sections.

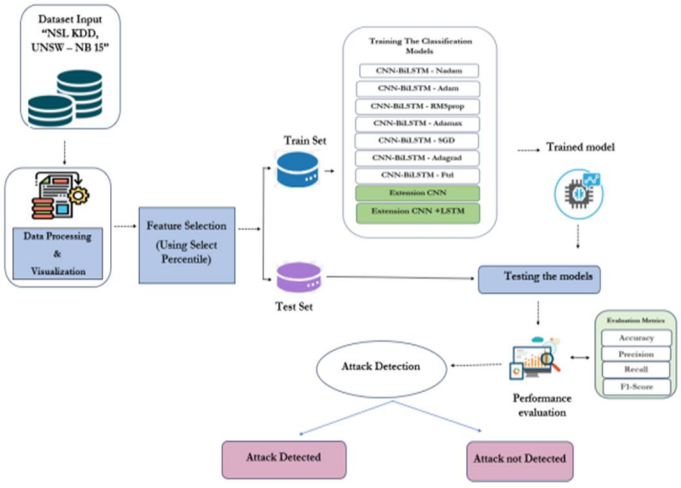


Fig 1 Proposed architecture

C. Collection of the Dataset

NSL-KDD [15] is a publicly available dataset, which has been created from the previous KDD cup99 dataset (Tavallaee et al., 2009). A statistical test conducted on the cup99 dataset raised several issues of concern that significantly affect the intrusion detection accuracy and leads to a biased assessment of AIDS (Tavallaee et al., 2009). The primary issue in the KDD data set is the enormous number of duplicate packets. Tavallaee et al. compared KDD training and test sets and discovered that around 78% and 75% of the network packets are replicated in both the training set and testing set (Tavallaee et al., 2009). This massive number of duplicate records in the training set would affect machine-learning techniques to favor normal records and therefore hinder them from acquiring unusual records that are normally more harmful to the computer system. Tavallaee et al. constructed the NSL-KDD dataset in 2009 from the KDD Cup'99 dataset in order to overcome the issues mentioned above by removing duplicated records (Tavallaee et al., 2009). The train dataset of NSL-KDD has 125,973 records and the test dataset consists of 22,544 records. The dimension of the NSL-KDD dataset is large enough to make it feasible to employ the entire NSL-KDD dataset without having to sample randomly. This has generated identical and similar results from different research studies. The NSL_KDD dataset has 22 training intrusion attacks and 41 features (i.e., attributes). In this dataset, 21 attributes refer to the connection itself and 19 attributes describe the nature of connections within the same host (Tavallaee et al., 2009).

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	...	dst_host_same_srv_rate	dst_host_diff_srv_rate	dst_hos
0	0	tcp	ftp_data	SF	491	0	0	0	0	0	...	0.17	0.03	
1	0	udp	other	SF	146	0	0	0	0	0	...	0.00	0.60	
2	0	tcp	private	S0	0	0	0	0	0	0	...	0.10	0.05	
3	0	tcp	http	SF	232	8153	0	0	0	0	...	1.00	0.00	
4	0	tcp	http	SF	199	420	0	0	0	0	...	1.00	0.00	

Fig 2 NSL KDD dataset

UNSW-NB15 is an intrusion network dataset. It has nine types of different attacks, comprising DoS, worms, Backdoors, and Fuzzers. The dataset has raw network packets. The training set records are 175,341 records and the testing set records are 82,332 records from the various types, attack and normal.

	id	dur	proto	service	state	spkts	dpkts	sbytes	dbytes	rate	...	ct_dst_sport_ltm	ct_dst_src_ltm	is_ftp_login	ct_ftp_cmd	ct_flw_http_mth
0	1	0.000011	udp	-	INT	2	0	496	0	90909.0902	...	1	2	0	0	
1	2	0.000008	udp	-	INT	2	0	1762	0	125000.0003	...	1	2	0	0	
2	3	0.000005	udp	-	INT	2	0	1068	0	200000.0051	...	1	3	0	0	
3	4	0.000006	udp	-	INT	2	0	900	0	166666.6608	...	1	3	0	0	
4	5	0.000010	udp	-	INT	2	0	2126	0	100000.0025	...	1	3	0	0	

5 rows × 45 columns

Fig 3 UNSW – NB 15 Dataset

D. Data Processing

Data processing is the process of converting raw data into useful information for businesses. Typically, data scientists process information, which involves collecting, organizing, cleaning, validating, analyzing, and transforming it into understandable forms like charts or reports. Data processing may be performed through three approaches i.e., manual, mechanical, and electronic. The objective is to enhance the value of data and make decision-making easier. This helps companies enhance their operations and make effective strategic decisions in a timely manner. Automated data processing technologies like computer programs programming have an important contribution to this. It has the ability to convert volumes of data, including big data, into insights that are meaningful for quality management and decision-making.

E. Feature Selection

Feature selection is the process of extracting the most reliable, non-redundant, and applicable features to employ in model building. Systematic reduction of dataset sizes is critical as the size and complexity of datasets keep increasing. The key aim of feature selection is to enhance the performance of a predictive model and minimize modeling computational expense.

Feature selection, which is part of the key element of feature engineering, refers to the method used to select the most significant features to be fed into machine learning algorithms. Feature selection methods are used to decrease the number of input variables by removing redundant or irrelevant features and condensing the list of features to include only those most significant to the machine learning model. The major advantages of conducting feature selection upfront, instead of allowing the machine learning algorithm to determine which features are most significant.

F. Algorithms

1) CNN (Convolutional Neural Network): An architecture of deep learning that is an expert in image processing, with convolutional layers to extract features and pooling layers for down sampling. CNNs perform well in image and pattern recognition applications and are therefore well suited to feature extraction from network traffic data. In this project, the isolated CNN is probably used as a baseline for comparison with more advanced architectures
2) CNN + BiLSTM with Nadam: This is a combination of Convolutional Neural Network (CNN) and Bidirectional Long-Short Term Memory (BiLSTM) layers. Bidirectional Long Short-Term Memory (BiLSTM) is a recurrent neural network (RNN) architecture that builds upon the functionality of typical LSTMs (Long Short-Term Memory networks) by processing sequences of information in both directions: forward and backward. BiLSTMs and LSTMs are very effective at processing sequential data
3) Nadam (Nesterov-accelerated Adaptive Moment Estimation): Nadam is a variation of the Adam optimizer. Nadam takes benefits from both adaptive moment estimation (Adam) and Nesterov accelerated gradient (NAG). NAG assists in speeding up the convergence by looking ahead for future gradients, and Adam adjusts learning rates for each parameter.
4) Nadam is used due to its capacity to deal with sparse gradients and noisy data, which can be common in network traffic data. CNN and BiLSTM are combined to enable the model to learn both spatial and temporal dependencies within the data.
5) CNN + BiLSTM with Adam: This model, much like the Nadam setup, uses CNN and BiLSTM layers together with
6) Adam (Adaptive Moment Estimation): Adam is a learning algorithm that calculates adaptive learning rates for every parameter by taking into account both the first-order momentum and the second-order (RMSprop) momentum of the gradients. The pairing with CNN and Bidirectional LSTM utilizes the best of each layer to identify intricate features and sequences.
7) CNN + BiLSTM with RMSprop: Yet another form of the hybrid model, using the RMSprop optimization algorithm.
8) RMSprop (Root Mean Square Propagation): RMSprop is an optimization algorithm that is designed to solve the issue of quickly decreasing learning rates in Adagrad by utilizing a moving average of squared gradients to normalize each parameter's learning rates.
9) RMSprop is chosen due to its stability in training deep networks with non-stationary objectives. When used with CNN and Bidirectional LSTM, it enables the model to learn efficiently spatial and temporal patterns in network traffic data.
10) CNN + BiLSTM with Adamax: The configuration uses Adamax as the optimization algorithm for the hybrid CNN + BiLSTM model. Adamax is a variation of the Adam optimizer, utilizing the infinity norm (maximum absolute value) of the running average of previous gradients. It makes the computation of the second-order momentum simpler. Its application with CNN and Bidirectional LSTM is used to achieve balance between computational cost and efficient feature learning.
11) CNN + BiLSTM with SGD (Stochastic Gradient Descent): A combination model utilizing Stochastic Gradient Descent as the optimizer. SGD is a simple optimizer that moves the model parameters in the negative direction of the gradient of the loss function by a factor of the learning rate. SGD, being a simple optimizer, is less computationally expensive and can be used as a baseline. Along with CNN and Bidirectional LSTM, it enables direct comparison to more advanced optimizers.
12) CNN + BiLSTM with Adagrad: This setup includes Adagrad as the optimization algorithm. Adagrad is an adaptive optimization algorithm that scales the learning rates of individual parameters by their past gradients, giving higher learning rates to less frequent parameters.
13) Adagrad adjusts learning rates per parameter based on past gradient data, and therefore it is applicable in sparse data scenarios. In the project, it can assist the model in learning efficiently from features of different importance
14) CNN + BiLSTM with Ftrl (Follow-the-regularized-Leader): A combination model employing Ftrl as the optimization algorithm, a blend of online learning and regularization. Ftrl has per-coordinate learning rates and adjusts to changing data distributions.
15) Ftrl (Follow-the-regularized-Leader): Ftrl is an optimization algorithm based on online learning combined with regularization. It has per-coordinate learning rates and employs an adaptive algorithm for updating weights.

IV. EXPERIMENTAL RESULTS

1) *Precision:* Precision evaluates the fraction of correctly classified instances or samples among the ones classified as positives. Thus, the formula to calculate the precision is given by:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

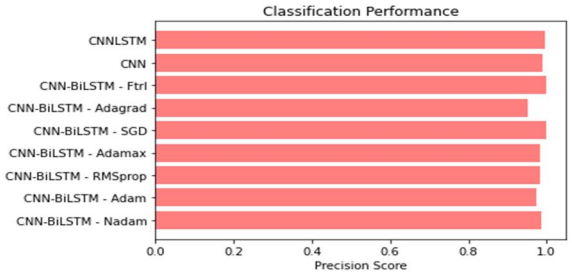


Fig 4 Precision comparison graph

2) Recall: Recall is a metric in machine learning that measures the ability of a model to identify all relevant instances of a particular class. It is the ratio of correctly predicted positive observations to the total actual positives, providing insights into a model's completeness in capturing instances of a given class.

$$Recall = \frac{TP}{TP + FN}$$

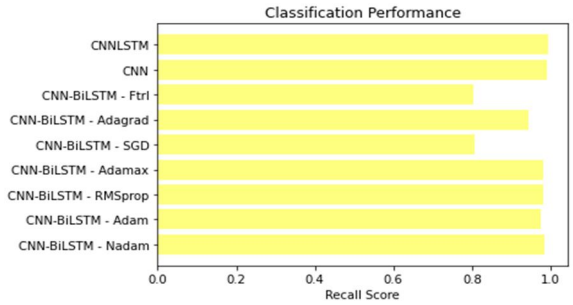


Fig 5 Recall comparison graph

3) Accuracy: Accuracy is the proportion of correct predictions in a classification task, measuring the overall correctness of a model's predictions.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

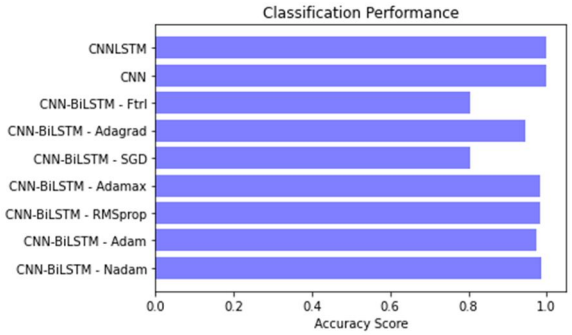


Fig 6 Accuracy graph

4) F1 Score: The F1 Score is the harmonic mean of precision and recall, offering a balanced measure that considers both false positives and false negatives, making it suitable for imbalanced datasets.

$$\text{F1 Score} = 2 * \frac{\text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} * 100$$

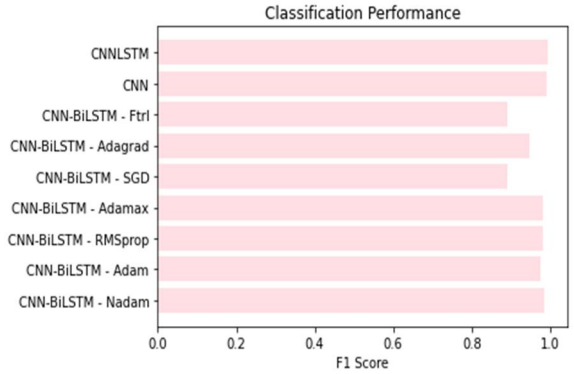


Fig 7 F1Score

ML Model	Accuracy	f1_score	Recall	Precision
CNN-BiLSTM - Nadam	0.986	0.986	0.986	0.986
CNN-BiLSTM - Adam	0.975	0.975	0.975	0.975
CNN-BiLSTM - RMSprop	0.982	0.982	0.982	0.982
CNN-BiLSTM - Adamax	0.983	0.983	0.983	0.983
CNN-BiLSTM - SGD	0.806	0.891	0.806	0.998
CNN-BiLSTM - Adagrad	0.945	0.947	0.945	0.952
CNN-BiLSTM - Ftrl	0.804	0.891	0.804	1.000
Extension CNN	1.000	0.990	0.990	0.990
Extension CNN +LSTM	1.000	0.995	0.995	0.99

Fig 8 Performance Evaluation

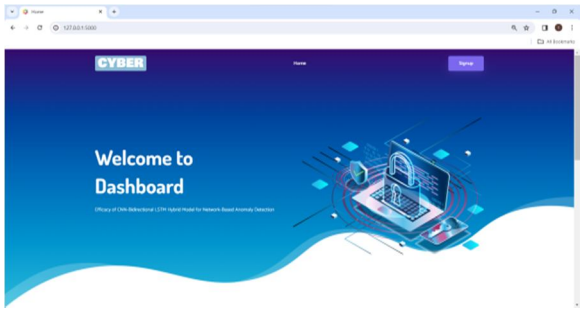


Fig 9 Home page

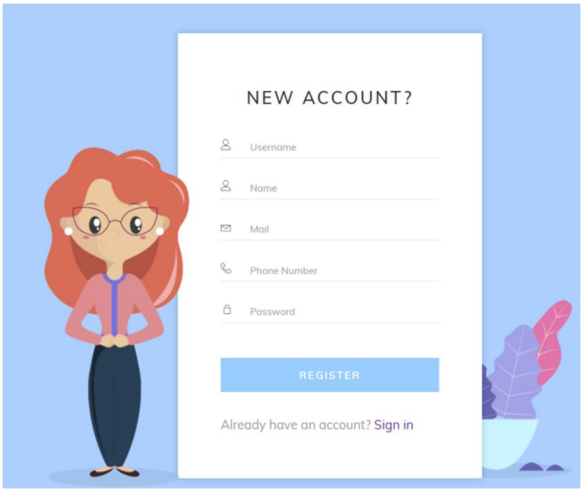


Fig 10 Signin page

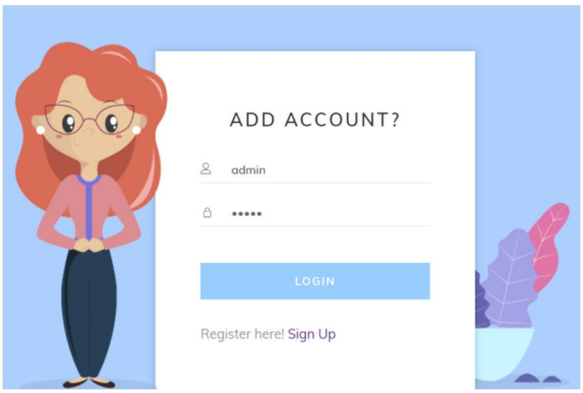


Fig 11 Login page

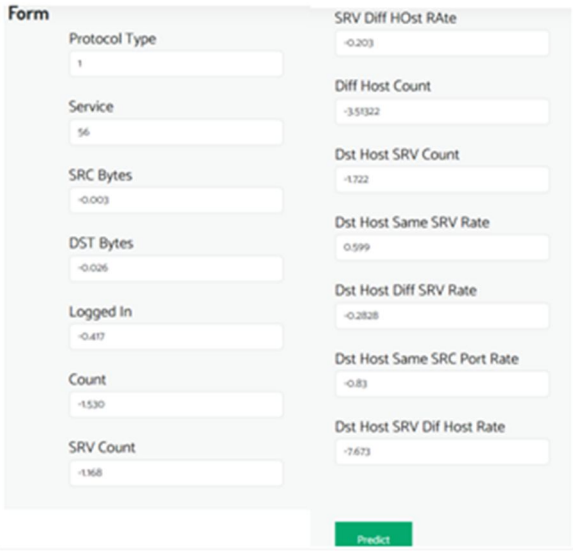


Fig 12 User input

Fig 13 Predict result for given input

V. CONCLUSION

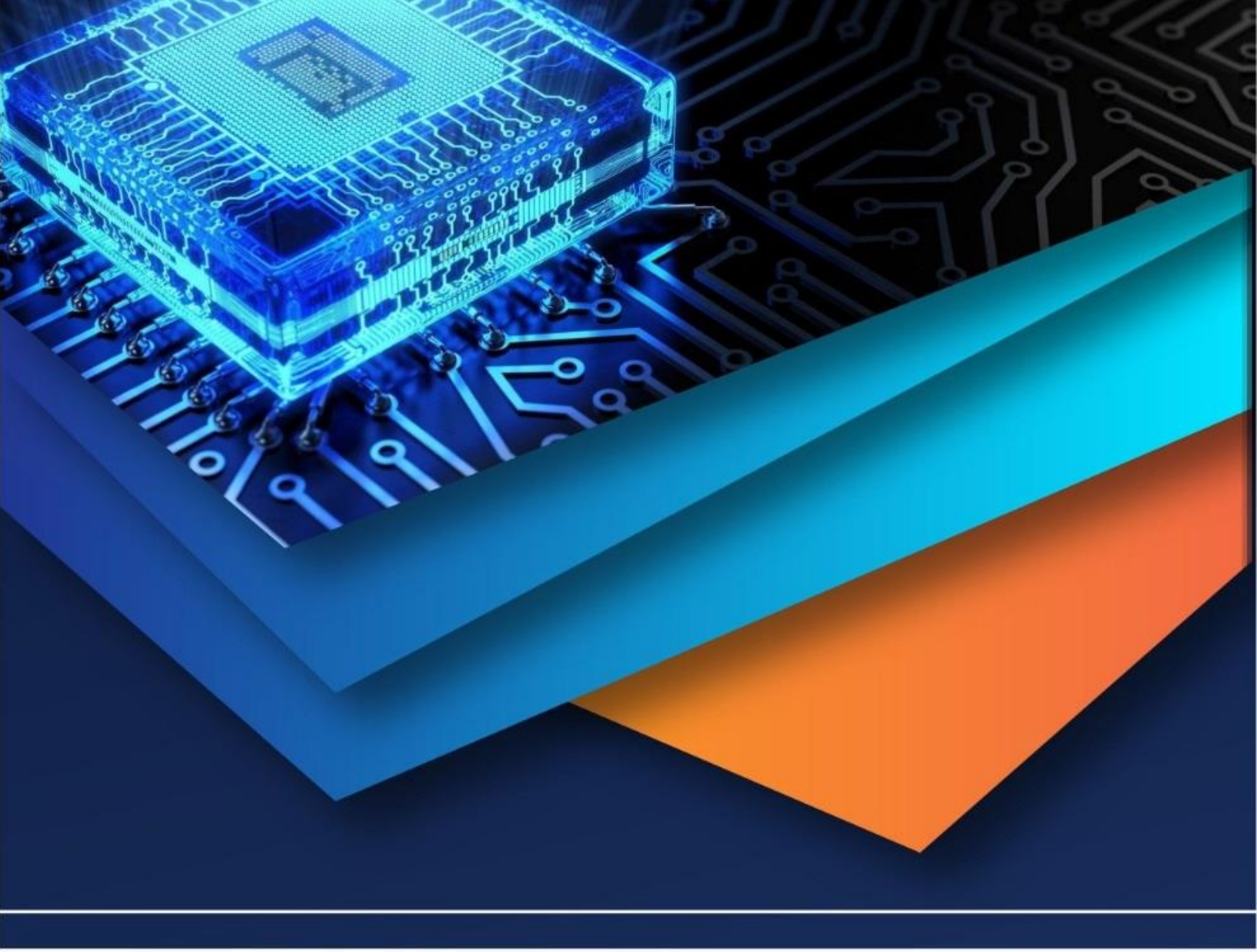
The investigation of different architectures of neural networks and optimization methods showed that the CNN + BiLSTM model [13, 14, 15], combined with a well-chosen optimizer, is the best solution for intrusion detection from network traffic data. Thorough evaluation metrics, such as accuracy, precision, recall, and F1 score, not only confirmed the strength of the selected model but also proved its ability to find an equilibrium between precise anomaly detection and not being too liberal with false positives. The project also highlights the sensitivity of the performance of the model to hyperparameters and the importance of careful tuning. Apart from algorithmic decisions, the theoretical aspects of choosing hyperparameters, such as architecture design and learning rates, are instrumental in realizing the best intrusion detection results. The algorithm significantly enhances the performance of the CNN Bi-LSTM model, showing outstanding accuracy in the detection of network abnormality. Front-end testing confirms its effectiveness in precise interpretation and analysis of feature values, highlighting its stability for real-world use. The results of the project have far-reaching implications for network security in the real world. The successful use of deep learning models for anomaly detection holds the promise to improve the resistance of computer networks against adaptive threats. The theoretical insight into model architecture and hyperparameter tuning contributes to a more comprehensive design of intrusion detection systems. This project breaks ground for further research and deployment in the field of cybersecurity, highlighting the real-world value of the project's results.

VI. FUTURE SCOPE

The project's findings offer meaningful insights and actionable recommendations to bolster network security. These insights provide valuable guidance for future advancements in cybersecurity practices [15].

REFERENCES

[1] N. Moustafa, J. Hu and J. Slay, "A holistic review of network anomaly detection systems: A comprehensive survey," Journal of Network and Computer Applications, vol. 128, p. 33–55, 2019.
[2] S. Samonas and D. Coss, "The CIA strikes back: Redefining confidentiality, integrity and availability in security.," Journal of Information System Security, vol. 10, 2014.
[3] Y. Fu, Y. Du, Z. Cao, Q. Li and W. Xiang, "A Deep Learning Model for Network Intrusion Detection with Imbalanced Data," Electronics, vol. 11, p. 898, 2022.
[4] K. Jiang, W. Wang, A. Wang and H. Wu, "Network intrusion detection combined hybrid sampling with deep hierarchical network," IEEE Access, vol. 8, p. 32464–32476, 2020.
[5] W. Xu, J. Jang-Jaccard, T. Liu, F. Sabrina and J. Kwak, "Improved Bidirectional GAN-Based Approach for Network Intrusion Detection Using One-Class Classifier," Computers, vol. 11, p. 85, 2022.
[6] L. Vu and Q. U. Nguyen, "Handling imbalanced data in intrusion detection systems using generative adversarial networks," Journal on Information Technologies & Communications, vol. 2020, p. 1–13, 2020.
[7] T. Acharya, I. Khatri, A. Annamalai and M. F. Chouikha, "Efficacy of Heterogeneous Ensemble Assisted Machine Learning Model for Binary and Multi-Class Network Intrusion Detection," in 2021 IEEE International Conference on Automatic Control & Intelligent Systems (I2CACIS), 2021.
[8] T. Acharya, I. Khatri, A. Annamalai and M. F. Chouikha, "Efficacy of Machine Learning-Based Classifiers for Binary and Multi-Class Network Intrusion Detection," in 2021 IEEE International Conference on Automatic Control & Intelligent Systems (I2CACIS), 2021.
[9] C. Yin, Y. Zhu, J. Fei and X. He, "A deep learning approach for intrusion detection using recurrent neural networks," Ieee Access, vol. 5, p. 21954– 21961, 2017.
[10] Z. Chen, C. K. Yeo, B. S. Lee and C. T. Lau, "Autoencoder-based network anomaly detection," in 2018 Wireless telecommunications symposium (WTS), 2018.
[11] M. Ganesh, A. Kumar and V. Pattabiraman, "Autoencoder Based Network Anomaly Detection," in 2020 IEEE International Conference on Technology, Engineering, Management for Societal impact using Marketing, Entrepreneurship and Talent (TEMSMET), 2020.
[12] W. Xu, J. Jang-Jaccard, A. Singh, Y. Wei and F. Sabrina, "Improving performance of autoencoder-based network anomaly detection on nsl-kdd dataset," IEEE Access, vol. 9, p. 140136–140146, 2021.
[13] J. Gao, "Network Intrusion Detection Method Combining CNN and BiLSTM in Cloud Computing Environment," Computational Intelligence and Neuroscience, vol. 2022, 2022.
[14] A. G. Salman, Y. Heryadi, E. Abdurahman and W. Suparta, "Single layer & multi-layer long short-term memory (LSTM) model with intermediate variables for weather forecasting," Procedia Computer Science, vol. 135, p. 89–98, 2018.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY