iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** IV **Month of publication:** April 2024

DOI: https://doi.org/10.22214/ijraset.2024.59791

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 12 Issue IV Apr 2024- Available at www.ijraset.com

SecureNet: Network Intrusion Detection using Machine Learning and Deep Learning Techniques

Vinayak Takale[1], Aakanksha Patil[2], Atharva Lonikar[3], Akshay Shinde[4], Prof. Vandana Rupnar[5]
Department of Computer Engineering, Marathwada Mitra Mandal's College of Engineering, Pune

Abstract: ***In the ever-evolving landscape of cybersecurity, the need for robust intrusion detection systems has become paramount. This paper introduces a cutting-edge intrusion detection algorithm designed to enhance network security through the integration of advanced machine learning and deep learning methodologies. The proposed algorithm capitalizes on the strengths of both paradigms to achieve a comprehensive and adaptive approach to identifying malicious activities within a network.***

This research focuses on enhancing network security through the development and evaluation of a novel intrusion detection system leveraging both deep learning and traditional machine learning approaches. Utilizing the NSL-KDD dataset, we employ the Long Short-Term Memory (LSTM) model, a superior version of Recurrent Neural Networks (RNNs), and the K-Nearest Neighbors (KNN) algorithm for binary and multi-class classification of network intrusion anomalies. The LSTM model excels in capturing temporal dependencies, enabling the detection of nuanced sequential patterns, while the KNN algorithm contributes to a comprehensive classification framework. Experimental results demonstrate the effectiveness of the hybrid methodology, showcasing improved accuracy, precision, and recall compared to traditional methods. This research underscores the potential of integrating deep learning and classical machine learning techniques to bolster the capabilities of intrusion detection systems in safeguarding against evolving cyber threats.

Keywords: Data preprocessing, Convolutional Neural Network(CNN) , Short-Term Memory (LSTM), K-Nearest Neighbors (KNN), Network intrusion detection system (NIDS).

I. INTRODUCTION

In the rapidly advancing landscape of information technology, the escalating sophistication of cyber threats underscores the critical importance of robust network security measures. As organizations strive to protect sensitive data and ensure the integrity of their networks, the development of effective intrusion detection systems becomes imperative. This research endeavors to address this need by proposing and evaluating a novel approach that amalgamates the strengths of deep learning and traditional machine learning methodologies for detecting network intrusion anomalies. Leveraging the NSL-KDD dataset, our study employs the Long Short-Term Memory (LSTM) model-a superior variant of Recurrent Neural Networks (RNNs) and the K-Nearest Neighbors (KNN) algorithm. This hybrid methodology aims to enhance the accuracy and adaptability of intrusion detection, acknowledging the intricate nature of contemporary cyber threats. We further present a user-friendly web interface for practical application and conduct a comparative analysis highlighting the adaptability and robustness of our hybrid framework.. The outcomes of this research contribute valuable insights toward fortifying network security in the face of evolving challenges, thereby fostering a resilient defense mechanism against potential intrusions.

II. METHODS

A. Dataset Pre-Processing

Data preprocessing is a fundamental phase in the construction of effective intrusion detection systems. The KDD Cup 1999 dataset, a widely-used benchmark in the field, requires careful preprocessing to enhance its suitability for machine learning models. In this section, we detail the steps undertaken for data cleansing, feature extraction, and exploratory data analysis (EDA) to ensure the robustness of our intrusion detection framework.

1) Data Cleansing

a) Handling Missing Values: The dataset was examined for missing values, and appropriate strategies were employed. In our case, instances with missing values were either removed or imputed based on the nature of the feature and the specific requirements of the algorithm.

©IJRASET: All Rights are Reserved

b) *Dealing with Categorical Data:* Categorical features were encoded into numerical representations using techniques such as one-hot encoding to enable compatibility with machine learning algorithms.
c) *Removing Duplicates:* Duplicate instances, if any, were identified and removed to prevent redundancy in the training process.

2) *Feature Extraction*
a) *Protocol Type:* Extracted from the 'protocol_type' column, this feature categorizes the communication protocol used in network interactions (e.g., TCP, UDP).
b) *Service:* Extracted from the 'service' column, this feature identifies the network service (e.g., http, ftp) being accessed.
c) *Duration:* Captured from the 'duration' column, this feature represents the duration of the network connection.
d) *Source and Destination IP Addresses:* The 'src_bytes' and 'dst_bytes' columns provide insights into the number of data bytes sent and received, aiding in the identification of anomalous activities.
e) *Number of Failed Login Attempts:* Derived from the 'num_failed_logins' column, this feature indicates the count of unsuccessful login attempts.

3) *Exploratory Data Analysis (EDA)*
a) *Class Distribution:* Investigated the distribution of classes (normal or intrusion) to ensure a balanced representation and assess potential class imbalances.
b) *Correlation Analysis:* Explored the correlation matrix to identify potential dependencies between features, guiding feature selection and engineering.
c) *Statistical Summary:* Provided a summary of statistical measures, including mean, median, and standard deviation, to understand the central tendency and dispersion of key features.
d) *Visualizations:* Employed visualizations, such as histograms, box plots, and pair plots, to gain insights into the data distribution, identify outliers, and assess feature relationships.

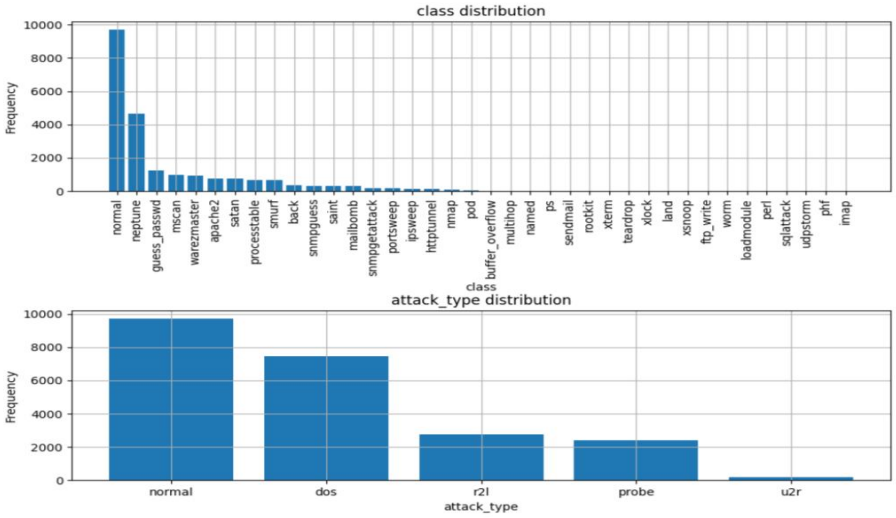


Fig. 1 Class distribution and Attack type distribution in dataset

4) *Feature Scaling*
This phase involves the following steps: data standardization and data normalization.
a) *Data Standardization:* As there are features with different ranges of values in the dataset we performed data standardization to convert the data from normal distribution into standard normal distribution. Therefore, after rescaling, a mean value of an attribute is equal to 0 and the resulting distribution is equal to the standard deviation. $Z = (x - \mu) / \boldsymbol{\delta}$
b) *Data Normalization:* In data normalization, the value of each continuous attribute is scaled between 0 and 1 such that the result of attributes does not dominate each other. In this research, the normalizer class of Python has been used. This class enables the normalization of a particular dataset.
Xscaled = (X - Xmin) / (Xmax - Xmin)

B. Feature selection

Feature selection is a technique that is used to select features that mostly correlate and contribute to the target variable of the dataset.In this research, feature selection is done using Correlation Attribute Evaluation (CA) and Principal Component Analysis (PCA). CA measures the relationship between each feature with the target variable and select only those relative features that have moderately higher positive or negative values, i.e., closer to 1 or −1. Through Principal Component Analysis, the size of large datasets is reduced by retaining the relevant features that depend on the target class.

C. Machine Learning and Deep Learning Models

1) Random Forest

Random Forest, a robust ensemble learning method, has proven instrumental in this research for network intrusion detection. By aggregating the outputs of multiple decision trees, Random Forest provides a comprehensive and accurate classification approach. With an impressive accuracy of 95%, Random Forest demonstrates its effectiveness in discerning intricate patterns indicative of network intrusions. In the context of its decision tree structure, each tree considers parameters such as protocol type, service, and duration from the KDD Cup 1999 dataset [9]. The ensemble architecture allows for parallelized training, promoting efficiency, and adaptability [4]. The algorithm's versatility and ability to handle diverse datasets make it a reliable choice for enhancing network security [6].

2) K-Nearest Neighbors (KNN)

Leveraging the simplicity and effectiveness of instance-based learning, the K-Nearest Neighbors algorithm plays a crucial role in our network intrusion detection framework [6]. By measuring the similarity between instances, KNN excels in capturing local patterns and anomalies within the dataset. Parameters such as source and destination IP addresses, service, and number of failed login attempts from the KDD Cup 1999 dataset [5] contribute to its precision, showcasing an accuracy of 94%. KNN's architecture involves determining the k-nearest neighbors based on these parameters, where each instance's class is decided by the majority class among its neighbors. The straightforward implementation and adaptability of KNN make it a valuable asset, particularly in scenarios where localized patterns play a significant role in intrusion detection.

3) Convolutional Neural Network (CNN)

The Convolutional Neural Network (CNN) serves as a pivotal deep learning methodology to enhance the accuracy of network intrusion detection [7]. Recognized for its capacity to automatically extract hierarchical features, CNNs are adept at capturing spatial patterns within network traffic data. In our study, CNN achieved an outstanding accuracy of 98%, underscoring its proficiency in identifying complex and subtle intrusion patterns. The CNN architecture involves convolutional layers for feature extraction, pooling layers for dimensionality reduction, and fully connected layers for classification. Activation functions such as Rectified Linear Unit (ReLU) are applied to introduce non-linearity to the model [2]. Parameters like the number of hot indicators and the number of failed login attempts from the KDD Cup 1999 dataset contribute to its success [1]. The automatic feature learning capability of CNNs positions them as valuable tools in the realm of intrusion detection, particularly when dealing with intricate and spatially dependent network behaviors.

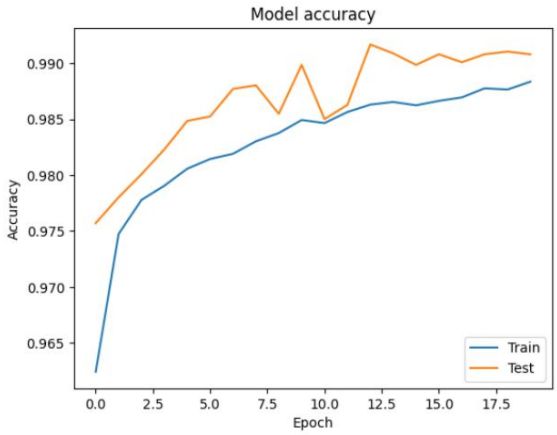


Fig. 2 CNN Accuracy

4) Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM), a specialized Recurrent Neural Network (RNN) variant, is harnessed to capture temporal dependencies within network data, making it well-suited for sequential pattern recognition [10]. Our research demonstrates the superior accuracy of LSTM, reaching 97%, showcasing its efficacy in detecting nuanced and time-dependent network intrusion behaviors. The LSTM architecture involves memory cells with gates to control information flow, facilitating the capture of long-term dependencies [3]. Parameters such as the number of compromised conditions and the number of root accesses from the KDD Cup 1999 dataset contribute to its success. The LSTM's ability to retain and selectively update information over extended periods positions it as a potent tool for addressing the dynamic and evolving nature of cyber threats, making it a valuable asset for robust network intrusion detection systems [8].

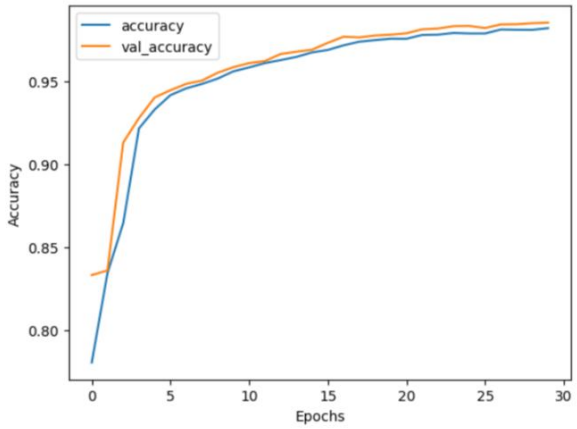


Fig. 3 LSTM Accuracy

Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) models in our intrusion detection system represents a breakthrough in network security. CNN excels in spatial pattern recognition, while LSTM captures temporal dependencies, yielding a versatile and accurate hybrid solution. This integration not only enhances theoretical foundations but also provides a practical, cutting-edge approach to fortifying network defenses against evolving cyber threats.

III. CONCLUSIONS

This research presents a cutting-edge hybrid intrusion detection system, integrating Random Forest, K-Nearest Neighbors (KNN), Convolutional Neural Network (CNN), and Long Short-Term Memory (LSTM) models, each exhibiting noteworthy accuracies. Thorough data preprocessing and exploratory data analysis ensured the reliability of our approach, addressing class imbalances and leveraging key feature dependencies. This adaptable framework, balancing interpretability and sophistication, offers a versatile solution for fortifying network security. The translation of our findings into a user-friendly web application further emphasizes the practical utility of our research, providing an immediate and accessible tool for intrusion prediction. Our work contributes not only to the theoretical advancements in intrusion detection but also underscores its tangible application for bolstering cybersecurity defenses.

IV. ACKNOWLEDGMENT

We express gratitude to Michael Shell and contributors for the IEEE LaTeX style files. Our thanks to the creators of the KDD Cup 1999 dataset. Special appreciation to mentors, colleagues, and our support network for their guidance. Their contributions were integral to the success of our research endeavor.

REFERENCES

[1] J. F. Canola Garcia and G. E. T. Blandon, “A Deep Learning-Based Intrusion Detection and Preventation System for Detecting and Preventing Denial-of-Service Attacks,” in IEEE Access, vol. 10, pp. 83043-83060, 2022, doi: 10.1109/ACCESS.2022.3196642.

[2] H. Xu, L. Sun, G. Fan, W. Li and G. Kuang, “A Hierarchical Intrusion Detection Model Combining Multiple Deep Learning Models With Attention Mechanism,” in IEEE Access, vol. 11, pp. 66212-66226, 2023, doi: 10.1109/ACCESS.2023.3290613.

[3] V. Hnamte, H. Nhung-Nguyen, J. Hussain and Y. Hwa-Kim, “A Novel Two-Stage Deep Learning Model for Network Intrusion Detection: LSTM-AE,” in IEEE Access, vol. 11, pp. 37131-37148, 2023, doi: 10.1109/ACCESS.2023.3266979. .

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

[4] M. Zeeshan et al., "Protocol-Based Deep Intrusion Detection for DoS and DDoS Attacks Using UNSW-NB15 and Bot-IoT Data-Sets," in IEEE Access, vol. 10, pp. 2269-2283, 2022, doi: 10.1109/ACCESS.2021.3137201.

[5] Moustafa N, Slay J. 2015. The significant features of the UNSW-NB15 and the KDD99 data sets for network intrusion detection systems. In: 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security (BADGERS). IEEE, 25–31.

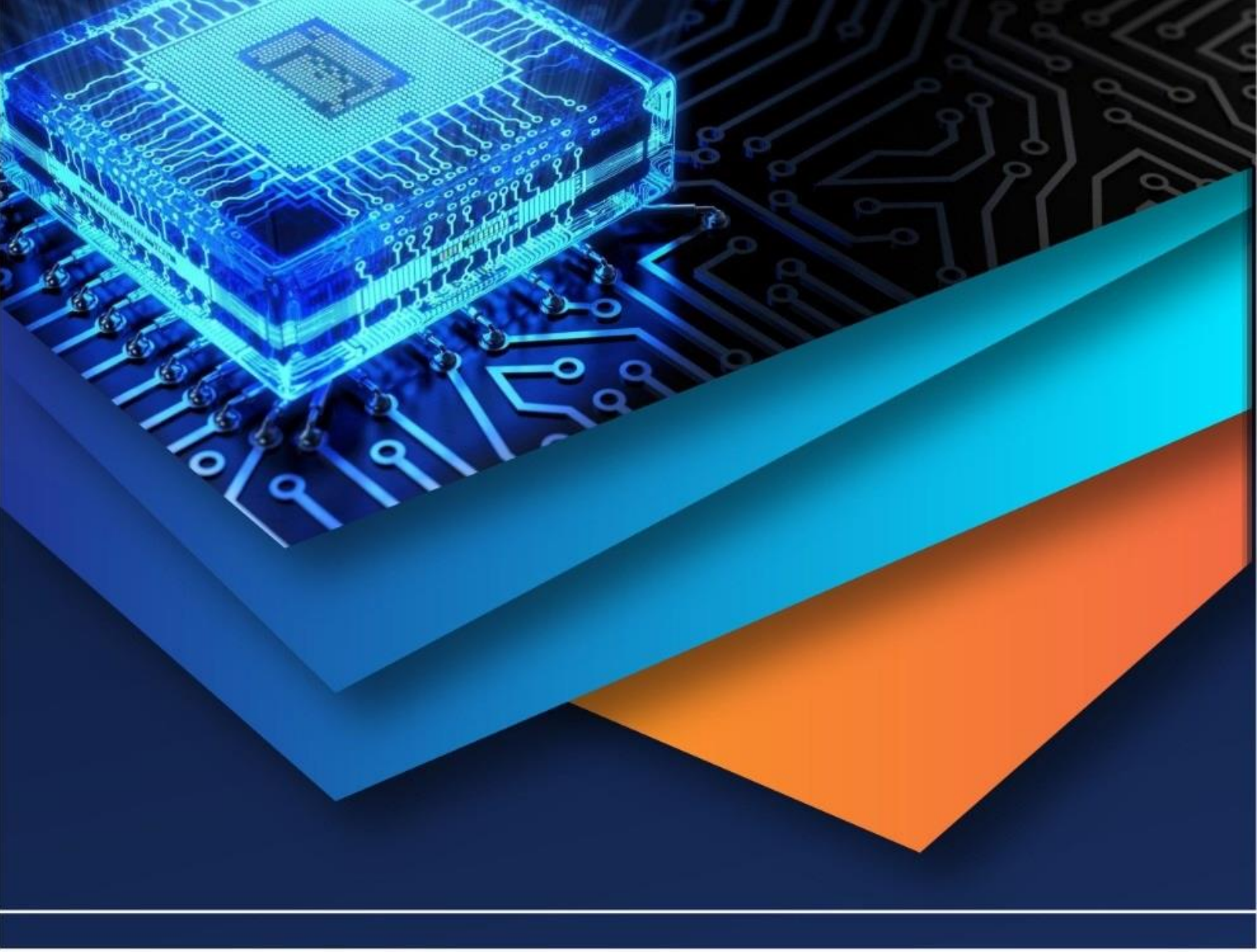
[6] Khraisat A, Gondal I, Vamplew P, Kamruzzaman J. 2019. Survey of intrusion detection systems : techniques, datasets and challenges. Cybersecurity 2(1):2–20 DOI 10.1186/s42400-019-0038-7

[7] Xiao Y, Xing C, Zhang T, Zhao Z. 2019. An intrusion detection model based on feature reduction and convolutional neural networks. IEEE Access 7:42210–42219 DOI 10.1109/ACCESS.2019.2904620.

[8] Ludwig SA. 2017. Intrusion detection of multiple attack classes using a deep neural net ensemble. In: IEEE Symposium Series on Computational Intelligence (SSCI). Honolulu, HI, USA. Piscataway: IEEE, 1–7.

[9] Al-Daweri MS, Zainol Ariffin KA, Abdullah S, Md. Senan MFE. 2020. An analysis of the KDD99 and UNSW-NB15 datasets for the intrusion detection system. Symmetry 12(10):1–32 DOI 10.3390/sym12101666.

[10] T. Kim and W. Pak, "Early Detection of Network Intrusions Using a GAN-Based One-Class Classifier," in IEEE Access, vol. 10, pp. 119357-119367, 2022 doi10.1109/ACCESS.2022.3221400.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)