International Journal of HRM and Organizational Behavior

www.ijhrmob.com

editor@ijhrmob.com

International Journal of
HRM and Organizational Behavior

ISSN 2454 - 5015

Data Balancing and CNN based Network Intrusion Detection System

NAKKALA MOUNIKA, Assistant Professor, Dept of CSE, Chirala Engineering College, Chirala,
nakkalamounika2121@gmail.com

RAVULA RAJA GOPAL, PG Student - MCA, Dept of MCA, Chirala Engineering College, Chirala,
rrgn1236@gmail.com

Abstract: The increasing threat of cyber attacks underscores the critical need for robust network security measures. This project emphasizes the development of a Network Intrusion Detection (NID) system, leveraging Convolutional Neural Networks (CNNs) to tackle imbalanced datasets and enhance classification accuracy. By addressing data imbalances through techniques like Random Over-Sampling (ROS), Synthetic Minority Oversampling TEchnique (SMOTE), and Adaptive Synthetic Sampling (ADASYN), the system achieves balanced classification across various attack types. Evaluation on benchmark datasets such as NSL-KDD and BoT-IoT demonstrates the system's effectiveness in accurately detecting and classifying network attacks. Building upon the base model's success, this study extends its approach by incorporating ensemble methods, notably combining CNN with Long Short-Term Memory (LSTM) networks. This hybrid ensemble model significantly improves accuracy, reaching an impressive 99%. By aggregating the predictions of multiple individual models, the ensemble approach enhances the system's robustness and overall performance. This research not only underscores the importance of efficient network security measures but also provides practical insights into leveraging advanced deep learning techniques for more effective intrusion detection systems.

Index Terms — Network Security, Data Balancing, Machine Learning, Deep Learning, Convolutional Neural Networks.

1. INTRODUCTION

The rapid advancement of cloud computing, Internet of Things (IoT) technologies, and wireless communication generations has ushered in an era of unprecedented connectivity [1]. With millions of users and devices interconnecting through these advanced technologies, the landscape becomes ripe for cyber-security threats [2]. Securing users' information and safeguarding IoT devices are paramount to ensuring the continuity of communication processes [3]. However, as cyber-security attackers exploit the interconnectedness of these systems, the need for robust Network Intrusion Detection (NID) systems becomes increasingly critical [4].

In response to evolving attack methods, modern NID systems must possess the capability to discern novel attacks, even those unseen during training [5]. While

ISSN 2454 - 5015

machine learning (ML)-based NID systems offer promising solutions, ML engineers encounter several challenges during implementation, particularly in handling imbalanced datasets [6], [7]. Imbalanced datasets can lead to high False Alarm Rates (FAR) on minority classes, compromising the overall effectiveness of the NID system [8].

To address these challenges, researchers have proposed various strategies for enhancing the performance of NID systems. One such strategy involves the integration of different data balancing techniques into ML models [9]. Random oversampling, a popular technique, involves replicating randomly selected samples from minority classes to balance the dataset [10]. Additionally, techniques like Synthetic Minority Oversampling TEchnique (SMOTE) generate new synthetic samples based on nearest neighbor information, effectively mitigating class imbalance [11]. Variants of SMOTE, such as Borderline-SMOTE, further improve classification accuracy by focusing on samples near the class boundary [12].

Moreover, ML engineers grapple with the complexity of raw input features in network traffic data, which can hinder the performance of NID models [13]. While classical ML algorithms demonstrate competence on certain datasets, they often struggle with the intricate feature space of network traffic data [14]. This limitation has prompted researchers to delve into the realm of deep learning (DL) for more effective feature extraction and classification [15].

DL algorithms, such as Variational Auto-Encoders (VAE) and Generative Adversarial Networks (GAN), offer novel approaches to data representation and synthesis [16], [17]. By learning deep representations of input features, DL models can effectively distinguish between different attack types, enhancing the overall performance of NID systems [18].

In light of these challenges and advancements, this paper aims to explore the efficacy of various data balancing techniques and DL algorithms in enhancing the performance of NID systems. By conducting empirical evaluations on benchmark datasets and real-world scenarios, we seek to provide insights into the practical implementation of these techniques and their impact on cyber-security. Through our research, we aim to contribute to the development of more robust and adaptive NID systems capable of mitigating emerging cyber threats in cloud computing and IoT environments.

2. LITERATURE SURVEY

Recent advancements in IoT technologies and cloud computing have led to the proliferation of interconnected devices, creating new opportunities for cyber-security threats [2]. In response, researchers have proposed innovative approaches to intrusion detection systems (IDS) that leverage deep learning (DL) techniques for enhanced detection accuracy [2], [3], [4]. For instance, Fatani et al. (2021) introduced an IoT intrusion detection system that utilizes deep learning in conjunction with enhanced transient search optimization for improved performance [2]. Similarly, Gupta et al. (2021) developed Lio-IDS, which addresses class imbalance in intrusion detection by employing LSTM networks and an improved one-vs-one technique [3]. These studies highlight the growing trend of integrating DL into

ISSN 2454 - 5015

IDS architectures to bolster security measures in IoT environments.

Jiang et al. (2020) proposed a hybrid sampling approach combined with deep hierarchical networks for network intrusion detection [4]. By leveraging both traditional sampling methods and deep learning techniques, their system demonstrates robust performance in detecting network anomalies [4]. This hybrid approach underscores the importance of integrating diverse methodologies to enhance the effectiveness of intrusion detection systems.

In the realm of imbalanced network traffic, Zhang et al. (2019) introduced an intrusion detection system based on convolutional neural networks (CNNs) [7]. Their system addresses the challenges posed by imbalanced datasets by leveraging the discriminative power of CNNs to accurately classify network traffic instances [7]. This approach showcases the potential of deep learning techniques in handling complex data distributions inherent in network intrusion detection tasks.

Moreover, Liu et al. (2021) proposed a fast network intrusion detection system that combines adaptive synthetic oversampling with the LightGBM algorithm [8]. By dynamically synthesizing minority class samples and leveraging a lightweight gradient boosting framework, their system achieves high detection performance while maintaining computational efficiency [8]. This research highlights the significance of optimizing both data preprocessing and algorithm selection to achieve effective intrusion detection in real-time scenarios.

In the context of ad-hoc networks, Huang and Lei (2020) introduced IGAN-IDS, an imbalanced generative adversarial network for intrusion detection [14]. By harnessing the power of generative adversarial networks (GANs), their system learns to generate synthetic samples of minority classes, thereby mitigating class imbalance and improving detection accuracy [14]. This approach showcases the potential of adversarial training techniques in addressing class imbalance challenges in intrusion detection systems deployed in dynamic network environments.

Furthermore, Elghalhoud et al. (2022) emphasized the importance of data balancing and hyper-parameter optimization for machine learning algorithms in securing IoT networks [15]. Their study underscores the need for comprehensive optimization strategies to ensure the robustness and reliability of intrusion detection systems deployed in IoT environments [15]. By integrating data balancing techniques with hyper-parameter tuning, researchers can effectively enhance the performance of IDS models in complex network scenarios.

In the domain of anomaly-based intrusion detection, Tama et al. (2019) proposed TSE-IDS, a two-stage classifier ensemble for intelligent anomaly detection [17]. By combining multiple classifiers at different stages of the detection pipeline, their system achieves superior detection accuracy and resilience against adversarial attacks [17]. This ensemble-based approach highlights the importance of leveraging diverse modeling techniques to enhance the robustness of intrusion detection systems.

Overall, the literature survey reveals a growing emphasis on leveraging deep learning techniques, hybrid sampling methods, ensemble learning

ISSN 2454 - 5015

approaches, and adversarial training strategies to address the challenges posed by class imbalance and complex data distributions in intrusion detection systems. By integrating these innovative methodologies, researchers can develop more effective and adaptive security solutions capable of mitigating emerging cyber threats in diverse network environments.

3. METHODOLOGY

a) Proposed Work:

The proposed work introduces a novel approach to Network Intrusion Detection (NID) by harnessing the power of Convolutional Neural Networks (CNNs) to address the challenges posed by imbalanced datasets. CNNs, renowned for their ability to extract hierarchical features from raw data, are well-suited for tasks such as image classification and, importantly, network attack classification. By leveraging CNNs, our system aims to enhance the classification accuracy of various attack types, including minority classes, thereby bolstering overall detection performance and network security.

In addition to CNNs, our system integrates various data balancing techniques such as Random Over-Sampling (ROS), Synthetic Minority Oversampling Technique (SMOTE)[4], and Adaptive Synthetic Sampling (ADASYN). These techniques are employed to mitigate the adverse effects of imbalanced datasets on model training, ensuring a more balanced representation of all attack types. Through this comprehensive approach, our proposed system endeavors to achieve improved accuracy and robustness in classifying network attacks, ultimately enhancing the effectiveness of NID systems in detecting and mitigating cyber threats.

b) System Architecture:

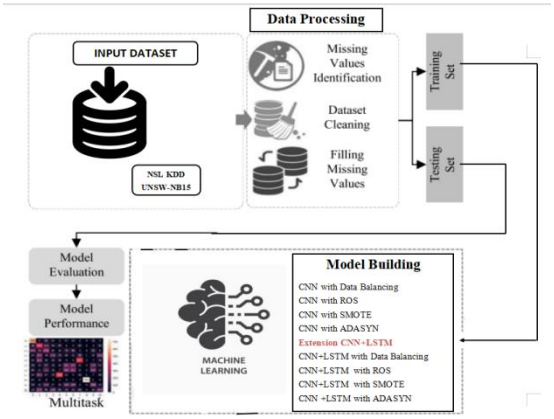

Fig 1 Proposed Architecture

The system architecture comprises several key components aimed at building an effective Network Intrusion Detection (NID) system. Initially, the input dataset consists of NSL KDD[11] and UNSW NB-15 datasets, which undergo preprocessing steps such as missing values identification, dataset cleaning, and filling missing values to ensure data integrity.

Subsequently, the processed dataset is split into training and testing sets to facilitate model development and evaluation. The model building phase involves the implementation of Convolutional Neural Networks (CNNs) augmented with data balancing techniques, including Random Over.

Sampling (ROS), Synthetic Minority Oversampling Technique (SMOTE)[4], and Adaptive Synthetic Sampling (ADASYN). Additionally, an extension to this approach incorporates a CNN+LSTM architecture for enhanced performance.

ISSN 2454 - 5015

Following model training, evaluation metrics are employed to assess the performance of each model variant, considering factors such as accuracy, precision, recall, and F1-score. Furthermore, the system integrates multitask capabilities to handle multiple intrusion detection objectives simultaneously, enhancing its versatility and applicability in real-world scenarios.

Overall, this system architecture leverages advanced deep learning techniques and data balancing strategies to develop a robust NID system capable of effectively detecting and mitigating network attacks across diverse datasets and scenarios.

c) Dataset:

The NSL-KDD dataset serves as a benchmarking dataset for evaluating the performance of machine learning algorithms in coping with high class imbalance in network intrusion detection tasks. This dataset is an enhanced version of the KDD'99 dataset, addressing issues such as redundant samples and easily predictable data instances. With an imbalance ratio of 1295:1, the NSL-KDD dataset presents a significant class imbalance challenge, where U2R attack traffic constitutes only 0.04% of the training dataset, while DoS traffic comprises over 36%.

	id	dur	proto	service	state	spkts	dpkts	sbytes	dbytes	rate	...	ct_
0	1	0.000011	udp	-	INT	2	0	496	0	90909.090200	...	
1	2	0.000008	udp	-	INT	2	0	1762	0	125000.000300	...	
2	3	0.000005	udp	-	INT	2	0	1068	0	200000.005100	...	
3	4	0.000006	udp	-	INT	2	0	900	0	166666.660800	...	
4	5	0.000010	udp	-	INT	2	0	2126	0	100000.002500	...	
...	...	...	...	...	...	...	...	...	...	...	...	...
82327	82328	0.000005	udp	-	INT	2	0	104	0	200000.005100	...	
82328	82329	1.106101	tcp	-	FIN	20	8	18062	354	24.410067	...	
82329	82330	0.000000	arp	-	INT	1	0	46	0	0.000000	...	
82330	82331	0.000000	arp	-	INT	1	0	46	0	0.000000	...	
82331	82332	0.000009	udp	-	INT	2	0	104	0	111111.107200	...	

Fig 2 BOT-IOT DATASET

To further challenge the generalizability of proposed intrusion detection systems, the NSL-KDD dataset includes a Test-21 dataset, where data samples predicted correctly by all 21 machine learning algorithms have been removed. This ensures a rigorous evaluation of the system's ability to detect novel and challenging attack types.

	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent
0	0	tcp	http	SF	181	5450	0	0	0
1	0	tcp	http	SF	239	486	0	0	0
2	0	tcp	http	SF	235	1337	0	0	0
3	0	tcp	http	SF	219	1337	0	0	0
4	0	tcp	http	SF	217	2032	0	0	0
...	...	...	...	...	...	...	...	...	...
494016	0	tcp	http	SF	310	1881	0	0	0
494017	0	tcp	http	SF	282	2286	0	0	0
494018	0	tcp	http	SF	203	1200	0	0	0
494019	0	tcp	http	SF	291	1200	0	0	0
494020	0	tcp	http	SF	219	1234	0	0	0

Fig 3 NSL KDD Dataset

In addition to the NSL-KDD dataset, the evaluation also extends to the BoT-IoT dataset, which provides a real-world representation of IoT network traffic. This dataset offers diverse attack scenarios and network conditions, allowing for comprehensive validation and testing of the proposed intrusion detection system. Overall, the NSL-KDD and BoT-IoT datasets collectively provide a robust foundation for assessing the efficacy and robustness of intrusion detection algorithms in diverse network environments.

d) Data Processing:

Data processing involves preparing the dataset for model training and evaluation, utilizing both pandas and Keras dataframes to manipulate and preprocess the data effectively.

Pandas Dataframe: Pandas dataframe is employed for initial data manipulation tasks, such as loading the dataset and performing exploratory data analysis. The purpose here is to gain insights into the dataset's

ISSN 2454 - 5015

structure, identify any missing values or inconsistencies, and understand the distribution of features and target variables.

Keras Dataframe: Keras dataframe is utilized for more advanced data preprocessing tasks specific to deep learning models. This includes converting categorical variables into numerical representations through one-hot encoding, scaling numerical features to a uniform range, and splitting the dataset into training and testing sets. The purpose of using Keras dataframe here is to prepare the data in a format suitable for feeding into neural network architectures.

Dropping Unwanted Columns: One crucial step in data processing involves dropping unwanted columns that do not contribute to the model's predictive power or introduce noise. These columns may include identifiers, timestamps, or irrelevant features that could hinder model performance. The purpose of dropping unwanted columns is to streamline the dataset, reducing dimensionality and improving computational efficiency during model training and inference.

e) Visualization:

Data visualization plays a crucial role in understanding the underlying patterns and relationships within the dataset. Seaborn and Matplotlib are powerful libraries in Python for creating informative visualizations. Seaborn offers high-level functions for creating attractive statistical plots, while Matplotlib provides more flexibility for customizing visualizations.

Through Seaborn and Matplotlib, we can create various types of plots such as histograms, scatter plots, box plots, and heatmaps to explore the distribution of features, identify outliers, detect correlations, and visualize model performance metrics. These visualizations aid in gaining insights into the data and informing decisions during the preprocessing and modeling stages.

f) Label Encoding:

Label encoding is a preprocessing technique used to convert categorical variables into numerical representations. The LabelEncoder class from the scikit-learn library is commonly employed for this task. It assigns a unique integer to each category within a categorical variable, thereby enabling machine learning algorithms to interpret these variables.

By applying LabelEncoder to categorical features, we ensure compatibility with algorithms that require numerical input. However, it's important to note that label encoding may introduce ordinal relationships between categories, which could impact model performance. Therefore, it's essential to use label encoding judiciously, especially with nominal categorical variables.

g) Feature Selection:

Feature selection is a critical step in machine learning to identify the most relevant features that contribute to the predictive power of the model. SelectPercentile with Mutual Info Classify is a feature selection method based on mutual information, which measures the dependency between variables.

By utilizing SelectPercentile, we can automatically select the top percentile of features with the highest

ISSN 2454 - 5015

mutual information scores, effectively reducing the dimensionality of the dataset while retaining the most informative features. This helps to mitigate the curse of dimensionality, improve model efficiency, and potentially enhance model generalization performance.

h) Training & Testing:

Splitting the dataset into training and testing sets is crucial for evaluating the performance of deep learning models on unseen data. Typically, a common practice is to allocate 80-20 or 70-30 proportions for training and testing, respectively. During training, the model learns patterns and relationships within the training data, iteratively adjusting its parameters to minimize the loss function. This process involves feeding batches of data into the model and updating its weights using optimization algorithms. Once trained, the model is evaluated on the testing set to assess its generalization ability and performance metrics like accuracy, precision, recall, and F1-score. This evaluation helps identify potential issues such as overfitting or underfitting and ensures the robustness of the model across various applications.

i) Algorithms:

CNN with Data Balancing: This algorithm employs a CNN architecture to classify network traffic data while addressing imbalanced datasets. Techniques such as undersampling or oversampling are applied to ensure a balanced representation of different attack types, enhancing the CNN model's ability to learn from minority classes.

CNN with ROS: Utilizing a CNN model trained on a dataset where the minority class is oversampled randomly, this algorithm achieves a more balanced distribution of classes. ROS involves creating synthetic instances of minority class data to match the frequency of the majority class, improving the CNN's ability to detect rare attack types.

CNN with SMOTE: Similar to ROS, SMOTE is applied to the CNN model's training dataset to generate synthetic instances of the minority class. By interpolating between existing instances, SMOTE creates a more balanced dataset, enhancing the CNN model's ability to recognize and classify less prevalent attack types.

CNN with ADASYN: ADASYN is employed to generate synthetic instances for the minority class based on their difficulty in learning regions. Focusing on areas where the CNN model might be less effective, ADASYN aids in improving the model's performance on minority class detection.

CNN+LSTM with Data Balancing: Combining CNN with LSTM for sequence learning, this algorithm addresses imbalanced datasets by using data balancing techniques in the training phase. The combination allows the model to learn both spatial and temporal features while handling class imbalances.

CNN+LSTM with ROS: Similar to the CNN+LSTM with Data Balancing, this approach uses ROS to balance the dataset before training the CNN+LSTM model, enhancing its capability to detect diverse network intrusions.

ISSN 2454 - 5015

CNN+LSTM with SMOTE: This algorithm involves applying SMOTE[4] to balance the dataset before training the CNN+LSTM model. By leveraging both CNN and LSTM layers and a balanced dataset, the model becomes more adept at capturing spatial and temporal patterns in network traffic, especially for less frequent attack types.

CNN+LSTM with ADASYN: ADASYN is utilized to balance the dataset for training the CNN+LSTM model, allowing it to effectively capture short-term and long-term dependencies in network traffic data, improving its accuracy in identifying various attack types.

4. EXPERIMENTAL RESULTS

Accuracy: The accuracy of a test is its ability to differentiate the patient and healthy cases correctly. To estimate the accuracy of a test, we should calculate the proportion of true positive and true negative in all evaluated cases. Mathematically, this can be stated as:

Accuracy = TP + TN TP + TN + FP + FN.

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

F1-Score: F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

$$\textbf{F1 Score} = \frac{2}{\left(\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}\right)}$$

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

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)

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

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.

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

ISSN 2454 - 5015

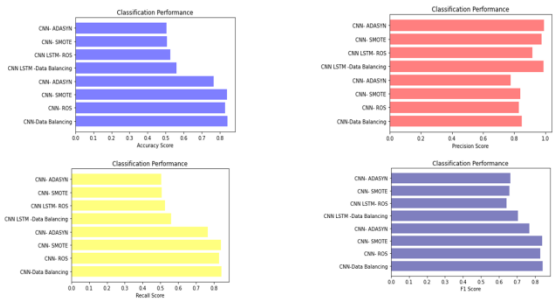

Fig 4 Comparison Graphs Of BOT-IOT Dataset

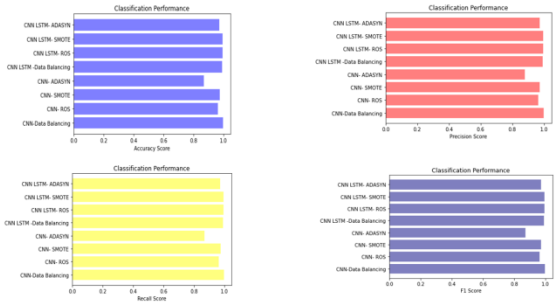

Fig 5 Comparison Graphs Of NSL-KDD Dataset

ML Model	Accuracy	f1_score	Recall	Precision
CNN-Data Balancing	0.995	0.996	0.995	0.997
CNN- ROS	0.963	0.963	0.963	0.963
CNN- SMOTE	0.973	0.973	0.973	0.973
CNN- ADASYN	0.869	0.873	0.869	0.880
Extension CNN LSTM -Data Balancing	0.990	0.991	0.990	0.993
Extension CNN LSTM- ROS	0.994	0.994	0.994	0.994
Extension CNN LSTM- SMOTE	0.994	0.994	0.994	0.994
Extension CNN LSTM- ADASYN	0.972	0.972	0.972	0.973

Fig 6 Performance Evaluation Table

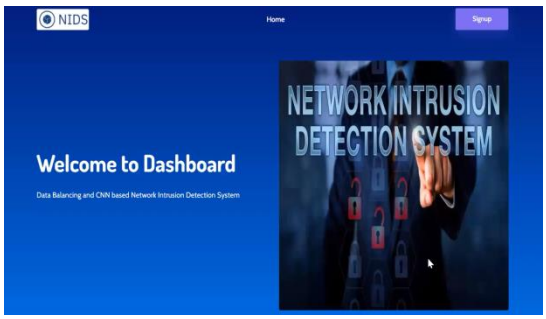

Fig 7 Home Page

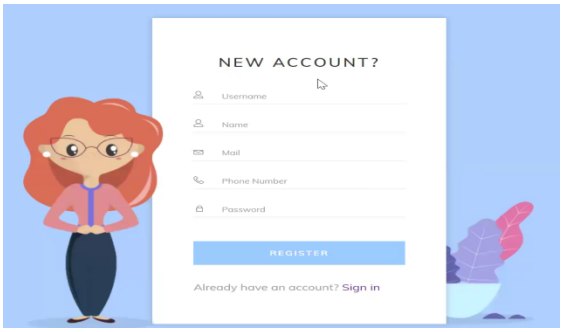

Fig 8 Registration Page

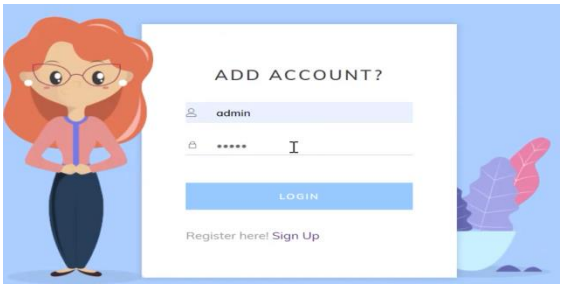

Fig 9 Login Page

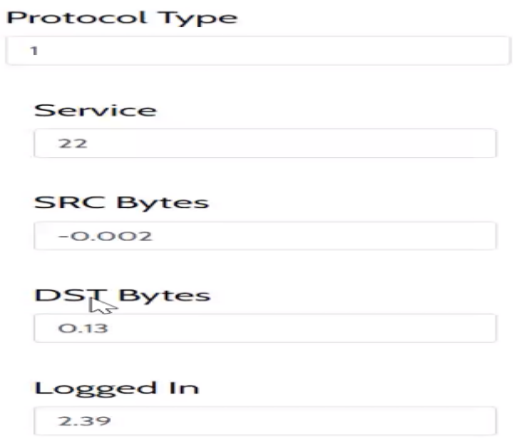

Fig 10 Upload Input Data

ISSN 2454 - 5015

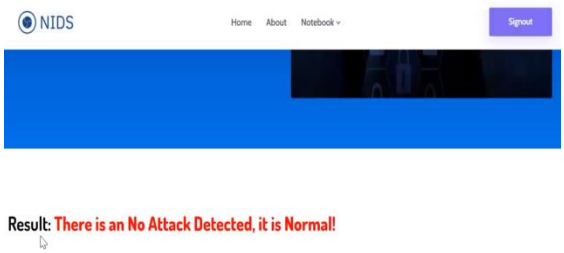

Fig 11 Final Outcome

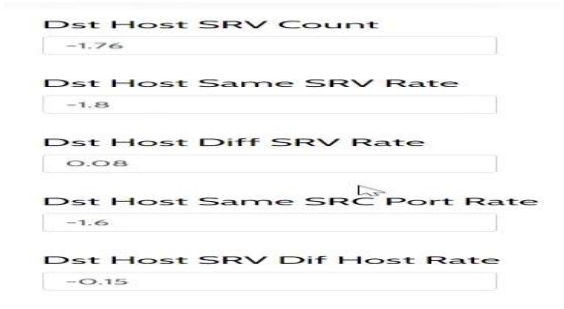

Fig 12 Upload Input Data

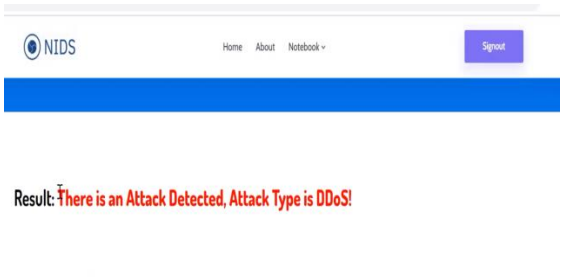

Fig 13 Predicted Results

Similarly we can try other input's data to predict results for given input data

5. CONCLUSION

The presented NID system, incorporating Convolutional Neural Networks (CNN) and addressing imbalanced datasets, exhibits robust performance in accurately classifying various types of network attacks. By employing proper data balancing techniques, the ML models effectively distinguish samples from minority classes without compromising performance on majority classes or overall system efficacy. Moreover, leveraging CNN for feature extraction yields significant performance improvements, highlighting the importance of advanced techniques in network intrusion detection. Comparative analysis against state-of-the-art systems demonstrates the superiority of the proposed approach, surpassing alternatives reliant on data balancing methods like ROS, SMOTE[4], and ADASYN. The extension model, employing a hybrid CNN+LSTM approach, achieves exceptional accuracy, further emphasizing its effectiveness in both data balancing and CNN-based intrusion detection. The integration of a user-friendly Flask interface with secure authentication enhances the system's usability during testing, streamlining data input and evaluation processes.

6. FUTURE SCOPE

Future research endeavors may focus on addressing the data imbalance issue through cost-sensitive learning techniques, enabling the NID system to adaptively adjust the misclassification costs based on class distributions. Additionally, exploration into advanced feature extraction methods beyond CNN, such as Graph Convolutional Networks (GCNs) or Transformer-based architectures, could further enhance the system's performance. Moreover, extending the system's capabilities to handle streaming data in real-time environments and integrating anomaly detection algorithms for proactive threat detection presents promising avenues

for future development. Furthermore, enhancing the scalability and efficiency of the Flask interface and incorporating advanced visualization tools for in-depth analysis of model performance could enrich the user experience and facilitate comprehensive system evaluation.

REFERENCES

[1] Y. Yang, K. Zheng, et al., "Improving the classification effectiveness of intrusion detection by using improved conditional variational autoencoder and deep neural network," Sensors, vol. 19, no. 11, 2019.

[2] A. Fatani, M. Abd Elaziz, et al., "Iot intrusion detection system using deep learning and enhanced transient search optimization," IEEE Access, vol. 9, pp. 123448–123464, 2021.

[3] N. Gupta, V. Jindal, and P. Bedi, "Lio-ids: Handling class imbalance using lstm and improved one-vs-one technique in intrusion detection system," Computer Networks, vol. 192, p. 108076, 2021.

[4] K. Jiang, W. Wang, A. Wang, and H. Wu, "Network intrusion detection combined hybrid sampling with deep hierarchical network," IEEE Access, vol. 8, pp. 32464–32476, 2020.

[5] R. Chapaneri and S. Shah, "Enhanced detection of imbalanced malicious network traffic with regularized generative adversarial networks," Journal of Network and Computer Applications, vol. 202, p. 103368, 2022.

[6] H. Ding et al., "Imbalanced data classification: A knn and generative adversarial networks-based hybrid approach for intrusion detection," Future Generation Computer Systems, vol. 131, pp. 240–254, 2022.

[7] X. Zhang, J. Ran, and J. Mi, "An intrusion detection system based on convolutional neural network for imbalanced network traffic," in IEEE 7th International Conference on Computer Science and Network Tech. (ICCSNT), pp. 456–460, 2019.

[8] J. Liu, Y. Gao, and F. Hu, "A fast network intrusion detection system using adaptive synthetic oversampling and lightgbm," Computers & Security, vol. 106, p. 102289, 2021.

[9] B. A. Tama and K. H. Rhee, "An in-depth experimental study of anomaly detection using gradient boosted machine," Neural Computing and Applications, vol. 31, pp. 955–965, 2017.

[10] Y. Yang, K. Zheng, B. Wu, Y. Yang, and X. Wang, "Network intrusion detection based on supervised adversarial variational auto-encoder with regularization," IEEE Access, vol. 8, pp. 42169–42184, 2020.

[11] M. Tavallaee et al., "A detailed analysis of the kdd cup 99 data set," in 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp. 1–6, 2009.

[12] N. Koroniotis, N. Moustafa, et al., "Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset," CoRR, vol. abs/1811.00701, 2018.

[13] A. Divekar et al., "Benchmarking datasets for anomaly-based network intrusion detection: Kdd cup

ISSN 2454 - 5015

99 alternatives," in IEEE 3rd Int. Conf. on Computing, Communication and Security (ICCCS), pp. 1–8, 2018.

[14] S. Huang and K. Lei, "Igan-ids: An imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks," Ad Hoc Networks, vol. 105, p. 102177, 2020.

[15] O. Elghalhoud, K. Naik, et al., "Data balancing and hyper-parameter optimization for machine learning algorithms for secure iot networks," In Proceedings of the 18th ACM Symposium on QoS and Security for Wireless and Mobile Networks (Q2SWinet '22), 2022.

[16] Z. Li, Qin, et al., "Intrusion detection using convolutional neural networks for representation learning," in Neural Information Processing, (Cham), pp. 858–866, Springer International Publishing, 2017.

[17] B. A. Tama, M. Comuzzi, and K.-H. Rhee, "Tse-ids: A two-stage classifier ensemble for intelligent anomaly-based intrusion detection system," IEEE Access, vol. 7, pp. 94497–94507, 2019.

Dataset Link:

Kdd-cup:
https://www.kaggle.com/datasets/kaggleprollc/nsl-kdd99-dataset

Bot-IoT:
https://www.kaggle.com/datasets/vigneshvenkateswaran/bot-iot-5-data