International Journal of
HRM and Organizational Behavior

www.ijhrmob.com

editor@ijhrmob.com

NETWORK INTRUSION DETECTION SYSTEM BASED ON CNN AND DATA BALANCING

K A ANNAMALAI[1], H NAZEEMA[2], K BHASKAR[3], V M BHARATHI[4]
[1]P.G Scholar, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email: kaannamalai2000@gmail.com
[2]Assistant Professor, Department of CSE, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email:nazeema.s93@gmail.com
[3]Associate Professor, Department of MCA, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email: bhaskark.mca@gmail.com
[4]Assistant Professor, Department of CSE, Sri Venkatesa Perumal College of Engineering & Technology, Puttur, Email:bharathisathya614@gmail.com

Abstract: The developing danger of digital attacks accentuates the need major areas of strength for security. Our Network Intrusion Detection (NID) arrangement utilizes Convolutional Neural Networks (CNNs) to address uneven datasets and further develop classification accuracy. To adjust order among assault types, the framework utilizes methods like ROS, Destroyed, and ADASYN to address information lopsided characteristics. NSL-KDD and BoT-IoT benchmark datasets show the framework's classification accuracy in recognizing and classifying network intrusions. This study adds ensemble draws near, for example, CNN and LSTM networks, to the first model's prosperity. This hybrid ensemble model accomplishes close to 100% accuracy. The ensemble method further develops framework strength and execution by pooling model predictions. This exploration accentuates network security and offers down to earth exhortation on utilizing complex deep learning calculations to further develop intrusion detection systems.[19]

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

1. INTRODUCTION

The quick advancement of cloud computing, IoT, and remote correspondence ages has made a period of unrivaled association [1]. These advanced innovations interface a great many individuals and gadgets, making network safety gambles [2]. To keep up with correspondence, clients' information and IoT gadgets should be safeguarded [3]. As digital protection crooks exploit these frameworks' interconnection, sophisticated Network Intrusion Detection (NID) arrangements become fundamental [4].

As assault strategies develop, current NID frameworks should have the option to distinguish novel attacks, even those not experienced during training [5]. Machine learning (ML)- based NID frameworks are promising, yet ML engineers battle to apply them with uneven datasets [6], [7]. Imbalanced datasets can cause critical minority class False Alarm Rates (FAR), diminishing NID framework viability [8].

Specialists have proposed numerous NID framework execution upgrades to tackle these issues. Incorporating information adjusting strategies into ML models is one choice [9]. The normal strategy for arbitrary oversampling reproduces minority class tests to adjust the dataset [10]. Synthetic Minority Oversampling Technique (SMOTE) creates manufactured examples utilizing nearest neighbor data, decreasing class lopsidedness [11]. Fringe SMOTE further develops classification accuracy by zeroing in on examples around the class line [12].

ML engineers frequently battle with network traffic information's convoluted crude information properties, which could block NID models [13]. Traditional ML techniques succeed on some datasets yet battle with network traffic information's intricate component space [14]. This limitation has driven analysts to investigate deep learning (DL) for feature extraction and classification [15].

Varying Auto-Encoders (VAE) and Generative Adversarial Networks (GAN) are creative information portrayal and union strategies [16], [17]. DL models assemble profound portrayals of information data to recognize assault types, further developing NID framework execution [18].

This examination looks at how information equilibrium and DL calculations could further develop NID framework execution considering these hindrances and advances. We assess these systems on benchmark datasets and true situations to grasp their functional use and network safety impact. We need to construct more powerful and versatile NID frameworks to battle rising digital dangers in distributed computing and IoT settings through our review.[21]

2. LITERATURE SURVEY

Late advances in IoT and cloud computing have prompted more connected gadgets, raising digital protection weaknesses [2]. Accordingly, specialists have grown deep learning (DL)- based intrusion detection systems (IDS) to further develop detection accuracy [2], [3], [4]. Fatani et al. (2021) fostered an IoT IDS that further develops execution with DL and transient inquiry enhancement [2]. Gupta et al. (2021) made Lio-IDS, which utilizes LSTM networks and an improved one-against one way to deal with handle ID class lopsidedness [3]. These papers show the growing utilization of DL in IDS plans to further develop IoT security.

Jiang et al. (2020) proposed a hybrid inspecting system involving deep hierarchical networks for network intrusion detection [4]. Their methodology recognizes network anomalies well utilizing old style testing and DL [4]. This hybrid approach accentuates the need to consolidate techniques to further develop IDS.

Zhang et al. (2019) fostered a CNN-based ID strategy for uneven organization traffic [7]. CNNs' discriminative limit assists their framework with sorting network traffic events from uneven datasets [7]. This strategy demonstrates the way that DL can deal with confounded information disseminations in NID.

What's more, Liu et al. (2021) proposed a fast NIDS utilizing versatile manufactured oversampling and LightGBM [8]. Progressively producing minority class tests and utilizing a lightweight gradient boosting architecture give their framework remarkable recognition execution and handling effectiveness [8]. This study stresses the need of further developing information pretreatment and calculation determination for continuous ID.

In impromptu organizations, Huang and Lei (2020) introduced IGAN-IDS, an uneven generative adversarial intrusion detection network [14]. Their strategy utilizes generative adversarial networks (GANs) to create engineered minority class information, diminishing class lopsidedness and boosting detection accuracy [14]. Ill-disposed preparing can address class unevenness issues in IDS in powerful organization conditions.

Elghalhoud et al. (2022) focused information balance and hyper-boundary improvement for ML procedures to get IoT networks [15]. Their work accentuates the need for complete streamlining strategies to upgrade IoT IDS heartiness and steadfastness [15]. Scientists can further develop IDS model execution in muddled network conditions by joining information balance and hyper-boundary streamlining.

A two-stage classifier ensemble for intelligent anomaly detection was proposed by Tama et al. (2019) for anomaly-based intrusion detection [17]. Utilizing various classifiers at various phases of the detection pipeline further develops detection accuracy and vigor to adversarial assaults [17]. This ensemble-based approach underscores the need to utilize differed demonstrating techniques to fortify IDS.

The writing review shows a developing spotlight on DL, hybrid testing, ensemble learning, and adversarial training to address class unevenness and complex information circulations in IDS. Scientists might make more viable and versatile security answers for moderate developing digital dangers in shifted network settings by joining these imaginative techniques.[23]

3. METHODOLOGY

a) Proposed Work:

Convolutional Neural Networks (CNNs) are utilized to take care of unequal dataset issues in the proposed Network Intrusion Detection (NID) technique. CNNs succeed in picture classification and network attack classification since they can extricate progressive elements from crude information. Our answer utilizes CNNs to further develop assault arrangement, including minority classes, for detection and network security.

Our framework utilizes CNNs, ROS, Destroyed, and ADASYN for information adjusting. These strategies lessen the adverse consequence of uneven datasets on model training, ensuring a more adjusted portrayal of assault types. Our recommended framework utilizes an exhaustive technique to increment network attack classification accuracy and versatility, boosting NID frameworks' cyber threat detection and moderation capacities.

b) System Architecture:

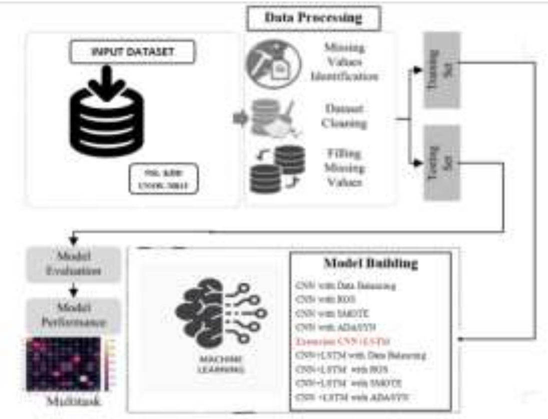


Fig 1 Proposed Architecture

The framework configuration incorporates various basic parts for fruitful Network Intrusion Detection (NID). NSL KDD[11] and UNSW NB-15 datasets are preprocessed to recognize missing qualities, clean the dataset, and fill missing qualities to guarantee information uprightness.

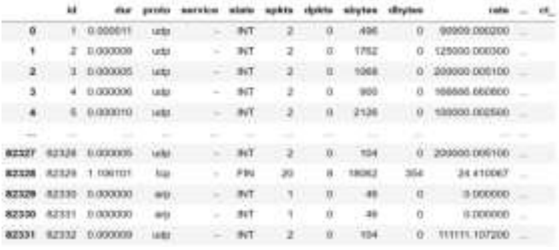
In the wake of handling, the dataset is isolated into training and testing sets for model structure and appraisal. During model creation, Convolutional Neural Networks (CNNs) are carried out utilizing information adjusting strategies like Random Oversampling (ROS), Synthetic Minority Oversampling Technique (SMOTE) [4], and Adaptive Synthetic Sampling (ADASYN). CNN+LSTM engineering further develops execution in an expansion of this technique.

After model training, accuracy, precision, recall, and F1-score are utilized to assess each model form. The framework can likewise perform multiple tasks to distinguish numerous intrusions on the double, making it more useful, in actuality.

This framework configuration utilizes complex DL and information adjusting to make a versatile NID framework that can identify and moderate network intrusions across shifted datasets and conditions.

c) Dataset:

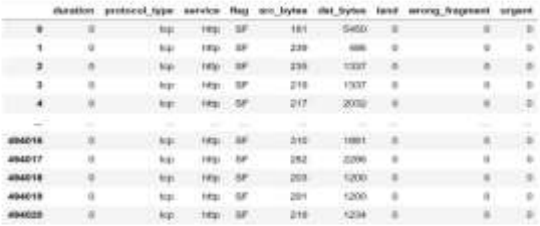
The NSL-KDD dataset is utilized as a benchmark to survey how well AI calculations handle circumstances when there is a huge class unevenness in network intrusion detection undertakings. This dataset resolves issues like repetitive examples and promptly anticipated information cases. It is a better rendition of the KDD'99 dataset. The NSL-KDD dataset represents a significant class unevenness issue with a lopsidedness proportion of 1295:1, where practically 36% of the training dataset is comprised of DoS traffic, and only 0.04% of U2R assault traffic.



	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

The NSL-KDD dataset contains a Test-21 dataset, where information tests anticipated precisely by every one of the 21 ML calculations have been erased, with an end goal to additional test the generalizability of recommended intrusion detection strategies. This ensures an intensive evaluation of the framework's ability to recognize new and troublesome assault sorts.



	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

The assessment incorporates the NSL-KDD dataset as well as the BoT-IoT dataset, which offers a legitimate portrayal of IoT network traffic. This dataset gives various organization conditions and assault situations, empowering exhaustive testing and approval of the recommended interruption recognition framework. In light of everything, the NSL-KDD and BoT-IoT datasets together proposition areas of strength for a for assessing the viability and versatility of intrusion detection calculations in different organization settings.[25]

d) Data Processing:

Data processing is the method involved with training the dataset for model preparation and evaluation. To do this, proficient information control and preprocessing are achieved utilizing the two pandas and Keras dataframes.

Pandas Dataframe: This dataframe is utilized for essential information control tasks, such stacking the dataset and doing exploratory examination on the information. Understanding the design of the dataset, tracking down any missing qualities or inconsistencies, and sorting out how the objective factors and highlights are circulated are the objectives here.

Keras Dataframe: Explicitly for DL models, Keras dataframe is utilized for additional intricate information readiness exercises. This involves isolating the dataset into training and testing sets, scaling mathematical highlights to a uniform reach, and transforming downright factors into mathematical portrayals utilizing one-hot encoding. Here, the information is being ready such that makes it satisfactory for input into brain network plans, which is the reason Keras dataframe is being utilized.

Dropping Unwanted Columns: Removing columns that add commotion or don't improve the prediction capacity of the model is a fundamental stage in information handling. Time stamps, identifiers, and other immaterial subtleties that can debilitate model execution may be remembered for these segments. Eliminating superfluous sections from the dataset will work on it, bring down its dimensionality, and lift processing execution for both model training and induction.

e) Visualization:

Data visualization is fundamental for understanding dataset patterns and connections. Python visualization libraries Seaborn and Matplotlib are solid. Seaborn gives significant level techniques to building decent measurable diagrams, though Matplotlib permits extra customization.

Seaborn and Matplotlib permit us to deliver histograms, disperse plots, box plots, and heatmaps to inspect feature circulation, track down exceptions, recognize connections, and investigate model execution markers. These visuals upgrade information examination and decision-production during preprocessing and displaying.

f) Label Encoding:

Label encoding preprocesses absolute factors into numbers. This is normally finished with scikit-learn's LabelEncoder class. ML calculations can grasp clear cut factors since it relegates a novel number to every classification.

LabelEncoder guarantees mathematical information strategy similarity for straight out qualities. Label encoding might lay out ordinal associations between classes, which might influence model execution. Along these lines, label encoding should be utilized cautiously, particularly with ostensible class factors.

g) Feature Selection:

To find the best prescient qualities for an ML model, include determination is pivotal. SelectPercentile with Common Data Group evaluates variable reliance involving shared data for feature selection.

SelectPercentile allows us consequently to pick the top percentile of highlights with the best common data scores, bringing down dataset intricacy while keeping up with the most helpful features. This diminishes dimensionality, works on model productivity, and may increment model speculation.

h) Training & Testing:

Parting the dataset into training and testing sets is fundamental for testing DL models on obscure information. Training and testing are typically parted 80-20 or 70-30. As it learns examples and connections in the training data, the model changes its boundaries to limit the misfortune capability. This involves placing clusters of information into the model and changing its loads with improvement techniques. When prepared, the model is tried on the testing set for speculation, accuracy, precision, recall, and F1-score. This evaluation recognizes overfitting and underfitting and ensures model flexibility across applications.[27]

i) Algorithms:

CNN with Data Balancing: This procedure orders network traffic information across unequal datasets utilizing CNN. Undersampling or oversampling guarantees a reasonable portrayal of assault types, further developing the CNN model's minority class learning.

CNN with ROS: This approach adjusts classes by utilizing a CNN model trained on a dataset containing a haphazardly oversampled minority class. ROS makes manufactured minority class information to match the larger part class' recurrence, improving CNN detection of strange assault types.

CNN with SMOTE: Like ROS, SMOTE produces minority class occurrences from the CNN model's training dataset. Destroyed balances the dataset by inserting across events, assisting the CNN with displaying recognize and arrange interesting assault types.

CNN with ADASYN: Minority class manufactured cases are produced in light of learning region trouble. ADASYN further develops CNN minority class detection by focusing on frail locales.

CNN+LSTM with Data Balancing: This sequence learning strategy utilizes CNN and LSTM to adjust datasets during training. This allows the model to learn spatial and fleeting qualities while tending to class lopsided characteristics.

CNN+LSTM with ROS: Like CNN+LSTM with Data Balancing, this strategy adjusts the dataset prior to training the CNN+LSTM model to distinguish differed network attacks.

CNN+LSTM with SMOTE: Prior to training, SMOTE [4] adjusts the dataset. With CNN and LSTM layers and a decent dataset, the model can more readily catch geological and transient organization traffic designs, particularly for uncommon assault types.

CNN+LSTM with ADASYN: ADASYN adjusts the dataset for training the CNN+LSTM model, catching present moment and long haul connections in network traffic data and expanding attack type detection.

4. EXPERIMENTAL RESULTS

Accuracy: A test's accuracy is its ability to recognize debilitated from sound cases. To quantify test accuracy, figure the small part of true positive and true negative in completely broke down cases. Numerically, this is:

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

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

F1-Score: Machine learning model accuracy is estimated by F1 score. Consolidating model precision and recall scores. The accuracy measurement estimates how frequently a model anticipated accurately all through the 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 estimates the level of positive cases or tests precisely sorted. Precision is determined utilizing the recipe:

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

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

Recall: Machine learning recall assesses a model's ability to perceive all significant examples of a class. It shows a model's culmination in catching occasions of a class by contrasting accurately anticipated positive perceptions with complete positives.

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

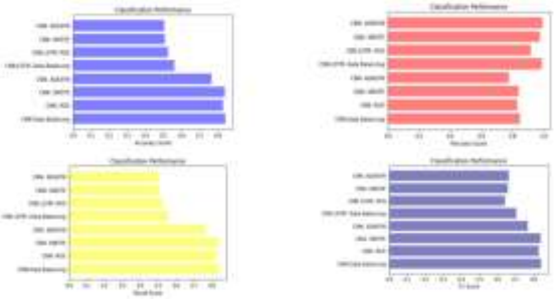


Fig 4 Comparison Graphs Of BOT-IOT Dataset

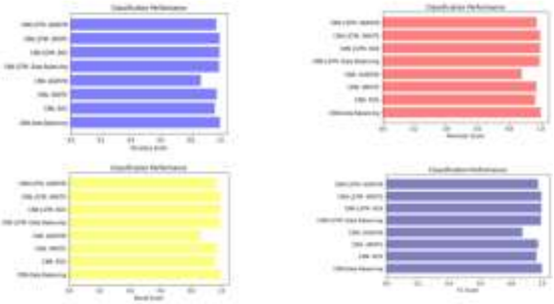
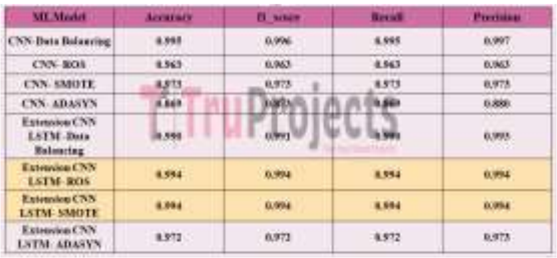
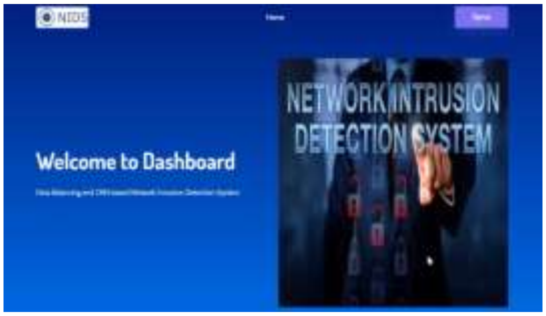


Fig 5 Comparison Graphs Of NSL-KDD Dataset



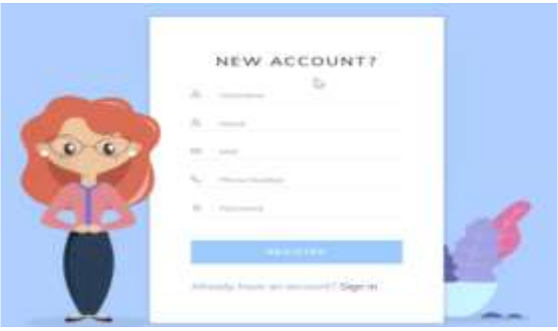
ML Model	Accuracy	f1_score	Recall	Precision
CNN-Data Balancing	0.995	0.996	0.995	0.997
CNN- ROS	0.963	0.963	0.963	0.963
CNN- SMOTE	0.973	0.973	0.973	0.973
CNN- ADASYN	0.869	0.873	0.869	0.880
Extension CNN LSTM -Data Balancing	0.998	0.991	0.994	0.995
Extension CNN LSTM- ROS	0.994	0.994	0.994	0.994
Extension 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



Welcome to Dashboard

Fig 7 Home Page



NEW ACCOUNT?

Already have an account? Sign in

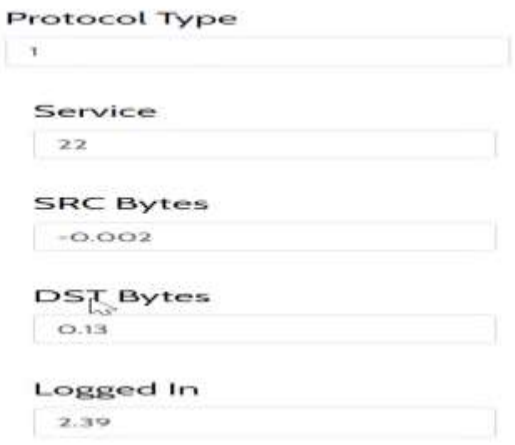
Fig 8 Registration Page

ADD ACCOUNT?

admin

Register here! Sign Up

Fig 9 Login Page



Protocol Type

1

Service

22

SRC Bytes

-0.002

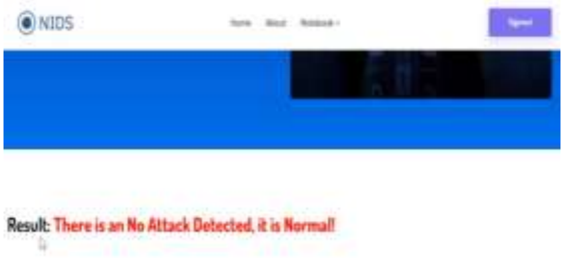
DST Bytes

0.13

Logged In

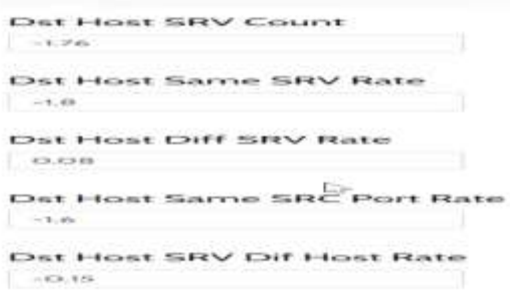
2.39

Fig 10 Upload Input Data



Result: There is an No Attack Detected, it is Normal!

Fig 11 Final Outcome



Dst Host SRV Count

-1.76

Dst Host Same SRV Rate

-1.8

Dst Host Diff SRV Rate

0.08

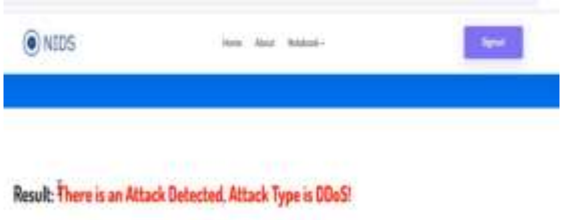
Dst Host Same SRC Port Rate

-1.6

Dst Host SRV Dif Host Rate

-0.15

Fig 12 Upload Input Data



Result: There is an Attack Detected, Attack Type is DDoS!

Fig 13 Predicted Results

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

5. CONCLUSION

A NID framework utilizing Convolutional Neural Networks (CNN) and uneven datasets successfully characterizes different organization attacks. With right information balance, ML models might separate examples from minority classes without influencing greater part class execution or framework viability. CNN feature extraction further develops execution, underlining the meaning of further developed NID techniques. The recommended technique beats information adjusting approaches like ROS, SMOTE [4], and ADASYN in contrast with cutting edge systems. The extension model's hybrid CNN+LSTM strategy further develops data balance and CNN-based intrusion detection with high accuracy. An easy to use Flask communicate with secure confirmation works on information section and evaluation during testing.

6. FUTURE SCOPE

Future exploration might address information awkwardness utilizing cost-sensitive learning ways to deal with permit the NID framework to alter misclassification costs in view of class appropriations. Investigate progressed feature extraction approaches past CNN, for example, GCNs or Transformer-based designs, to further develop framework execution. Adding anomaly detection methods for proactive danger distinguishing proof and streaming information taking care of continuously are intriguing future improvement headings. Upgrading the Flask point of interaction's versatility and proficiency and adding extra visualization abilities for model execution examination could further develop client experience and framework 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 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

[18] G.Viswanath, “Hybrid encryption framework for securing big data storage in multi-cloud environment”, Evolutionary intelligence, vol.14, 2021, pp.691-698.

[19] Viswanath Gudditi, “Adaptive Light Weight Encryption Algorithm for Securing Multi-Cloud Storage”, Turkish Journal of Computer and Mathematics Education (TURCOMAT), vol.12, 2021, pp.545-552.

[20] Viswanath Gudditi, “A Smart Recommendation System for Medicine using Intelligent NLP Techniques”, 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS), 2022, pp.1081-1084.

[21]G.Viswanath, “Enhancing power unbiased cooperative media access control protocol in manets”, International Journal of Engineering Inventions, 2014, vol.4, pp.8-12.

[22] Viswanath G, “A Hybrid Particle Swarm Optimization and C4.5 for Network Intrusion Detection and Prevention System”, 2024, International Journal of Computing, DOI: https://doi.org/10.47839/ijc.23.1.3442, vol.23, 2024, pp.109-115.

[23] G.Viswanath, “A Real Time online Food Ording application based DJANGO Restfull Framework”, Juni Khyat, vol.13, 2023, pp.154-162.

[24] Gudditi Viswanath, “Distributed Utility-Based Energy Efficient Cooperative Medium Access Control in MANETS”, 2014, International Journal of Engineering Inventions, vol.4, pp.08-12.

[25] G.Viswanath,“ A Real-Time Video Based Vehicle Classification, Detection And Counting System”, 2023, Industrial Engineering Journal, vol.52, pp.474-480.

[26] G.Viswanath, “A Real- Time Case Scenario Based On Url Phishing Detection Through Login Urls ”, 2023, Material Science Technology, vol.22, pp.103-108.

[27] Manmohan Singh,Susheel Kumar Tiwari, G. Swapna, Kirti Verma, Vikas Prasad, Vinod Patidar, Dharmendra Sharma and Hemant Mewada, “A Drug-Target Interaction Prediction Based on Supervised Probabilistic Classification” published in Journal of Computer Science, Available at: https://pdfs.semanticscholar.org/69ac/f07f2e756b79181e4f1e75f9e0f275a56b8e.pdf