International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 26 August 2023 **Revised:** 20 October 2023 **Accepted:** 02 November 2023*

An Effective and Efficient Intrusion Detection System of Network Attacks Using Stacked CNN and Voting Technique

[1*]Swati Mirlekar, [2]Dr. Komal Prasad Chourasia, [3]Dr. Bharti Chourasia
[1*]Assistant Professor, Department of Electronics & Communication Engineering,
St. Vincent Pallotti College of Engineering & Technology, Nagpur, Maharashtra
swati.mirlekar@gmail.com
[2]Associate Professor, Department of Electronics & Communication Engineering,
RKDF Institute of Science and Technology Bhopal, M.P
komal44@gmail.com
[3]Head of the Department, Department of Electronics and Communications Engineering,
Sarvepalli Radhakrishnan University (SRKU) Bhopal
chourasia3012@gmail.com

Abstract: IDS are crucial to network security because they can identify malicious activity and halt it in its tracks. Network intrusion data is often masked by a sea of benign data, making it difficult to train a model or perform a detection with a high FPR. This is because networks are inherently dynamic and change over time. In this research, we offer a ML & DL model-based method to ID, and we demonstrate how to deal with the issue of data imbalance by using a hybrid sampling technique. Conventional firewalls and data encryption technologies are unable to provide the level of security required by current networks. As a result, IDSs have been endorsed for use against network threats. Recent mainstream ID approaches have benefited from ML, but they have low detection rates & need a lot of feature engineering to be truly useful. Using layered CNN and Voting classifier (XGBoost and LGBM), this study introduces ML-DL-NIDS to address the issue of subpar detection precision. Using a publicly available NSL-KDD & UNSW-15 benchmark datasets for network intrusion detection, we find that this model outperforms competing methods according to accuracy and F1-score obtained from experimental evaluations.

Keywords: *ID, Machine Learning, DL, Stacked CNN; LGBM, XGBoost, Voting, NSL-KDD; Attention Mechanism; UNSW15.*

I. INTRODUCTION

U
SING the network, we may transport data easily, but doing so can expose us to numerous security risks. To protect ourselves from these risks, we utilise cyber security. Anti-virus software, firewalls, and other cyber security measures guard against unauthorised intrusions into networks. But they are not powerful enough to recognise a novel kind of attack. IDS is used to increase network security. IDS is utilized to identify, track, & examine any network vulnerabilities in both software and hardware[1]. An IDS is a system that keeps tabs on a network in order to spot any untoward happenings. There have been remarkable developments in the use of machine learning in areas such as healthcare, autonomous vehicles, fraud detection, personalization, entertainment, and robotics in recent years. The field of cybersecurity has profited from this advancement as well. Signature-based detection (or "misuse detection") and anomaly detection are the two primary types of IDSs used today. During signature-based detection, the IDS compares incoming data to known attack signatures. Popular programs like Snort and Suricata have helped spread this method, but it has a significant drawback: it can only detect threats that have already been described in a database.[2]. Anomaly detection, on the other hand, constructs a model of the system's normal behaviour and then searches for outliers in the data being watched. This method can thus detect previously undisclosed threats, but it also generates a high volume of false alarms. Extensive research into anomaly-based IDSs has been conducted over the past two decades. threats are getting more numerous and diversified, therefore their capacity to detect unknown threats is more important.

People and businesses face difficulties as a result of this. Given this context, attack detection methods need to be smarter and more powerful than ever before to fend off hackers' attacks, which are themselves constantly evolving. The majority of today's methods for identifying suspicious activity (threats) rely on event analysis with a set of predetermined rules. [3]. However, they are also constrained by factors like the absence of data on real attacks and the monetary losses from incorrectly identifying security breaches. Because of this, they are reserved for the only purpose of automatically collecting and analysing various security event information in order to evaluate threats. Because of these benefits, deep learning has emerged as the method of choice for detecting intrusions.

A. The motivation of the research

Since new technologies are constantly being developed and network architecture is constantly changing, today's networks are not completely safe. Multiple levels of security must be designed securely in order to meet these issues, i.e., a suitable defense-in-depth architecture must be put in place. Network IDS is one of these security levels. An IDS aids in alerting whether a sophisticated attack is currently underway[4]. As an alternative, if a previous attack was made and by whom, suggesting that it also aids in identifying the enemy and its actions. To stay current and be able to identify new assaults, intrusion detection systems must be regularly improved. The development of highly effective IDS still faces difficulties despite a fact that many studies have been done to advance a field.

B. Research contribution

- To develop an effective and efficient system for capable to identify network intrusion.
- To collect dataset of network traffic, encompassing both normal and malicious activities, for model training and evaluation.
- The raw data must be transformed into a form that can be easily processed by the chosen model architecture.
- To determine which models are best for network intrusion detection.
- To evaluate a model's detection efficiency using appropriate performance metrics on a test dataset.

The following is an overview of the investigation. Some recent studies on NID prevention are discussed in Section II. Section III includes more information about the study's research methodologies. Details about the dataset, the experiments, and the statistical analysis of the dataset are provided in Section IV. The investigation's findings are provided in Section V.

II. LITERATURE REVIEW

In the recent decade, researchers have developed a plethora of intrusion detection systems. These ranged from network-based to those meant to function in tandem with existing host-based IDS software. These systems are examples of hybrid solutions that incorporate HIDS/NIDS with signature- & anomaly-based approaches. As an example of an intelligence system, intrusion detection systems frequently employ computational intelligence. In order to build a reliable IDS, it is necessary to use both classic mathematical processes & analytical methodologies, as well as soft computing techniques. Examining how NIDS could fit into the current system is the focus of this study. In this paper, we first examine existing methods for detecting intrusions using machine learning, deep learning, and shallow learning. In this section, we'll go over what's already been written on the various DL methods that have been put to use in the field of science[5].

V. Sujatha [2023] Reinforcement learning strategies, such Q-learning and deep feed-forward neural networks, form the basis of modern ID solutions for today's networks. The suggested Deep Q-Learning (DQL) model uses an automated trial-and-error method to improve its detection abilities over time and find new types of network intrusions. The proposed model outperforms existing self-taught learning models by a wide margin, with precision of 92.8%, accuracy of 91.4%, and recall rate of 90.2%. Our results demonstrate that our proposed DQL outperforms other machine learning algorithms in its classification of intrusion kinds, and this is supported by experimental evidence[6].

Xiuye Yin [2022] develops a model for intrusion detection analysis networks using multi-scale convolutional neural networks (M-CNNs). To improve the model's local feature extraction performance, we incorporate the models of long-term and short-term memory networks into M-CNN. In addition, layers for batch normalization & global average pooling are added to the network to ensure that data is distributed uniformly across all layers, cut down on model training time and gradient calculation, and boost the network's overall performance. The simulation experiment shows that the M-CNN ID model outperforms the baseline on the KDDcup99 data set. A detection model has a precision of 93.90 and a recall of 93.59[7].

A. Lakshmanarao [2020] for IDS, the authors present three methods of feature selection, then apply ML and DL. To find the most relevant characteristics, we combined two datasets and employed an ANOVA F-value based method, an impuritybased feature selection, & a mutual information-based procedure. Finally, using two datasets, we applied three distinct ML techniques (K-NN, DT, LR, and DL Feed Forward Neural Networks), attaining an overall accuracy of 99.9% and a feed forward neural network accuracy of 88%. Our model outperformed modern methods, as evidenced by the findings[8].

A K M Mashuqur [2020] suggested model utilizes machine learning models to construct the IDS. This paper briefly discusses some alternative machine learning models and how they compare to the proposed model. These alternative models are AdaBoost, XGBoost, Gaussian Naive Bayes, Random Forest, & LGB. We tested the models on a NSL KDD dataset & found that our proposed model achieves an 11% improvement in accuracy over the competition [23].

Muhammad Ahmad Faraz [2020] This study proposes a statistical strategy as an alternative to the failed attempts at intrusion detection made by more conventional means. To identify the network attacks, a softmax classifier is used once features have been retrieved & selected utilizing a multilayer CNN. The research also makes use of two other popular ID datasets, NSL-KDD and KDDCUP'99. Accuracy, recall, F1-score, and precision are utilized as performance indicators to gauge how well the suggested model works. A testing results shows that the proposed method outperformed modern IDSs with an accuracy of 99% [24].

K. Singh [2019] It was shown that the SPELM method outperformed the DBN approach when used as a machine learning classifier on the NSL KDD dataset. There are four million records in the NSL- KDD dataset, with 40% utilized for training and 60% used for testing in order to determine whether algorithm is more effective. An experiment conducted by a scholar contrasted the computational time required by the existing DBN method with the new SPELM Algorithm on a basis of accuracy, precision, & recall. When compared to a DBN algorithm, the outcomes demonstrate that SPELM performs better. Its accuracy is 93.20%, while DBN's is only 52.8%; SPELM's precision is 69.492%, while DBN's is only 66.8368%; and SPELM's computational time is only 90.8 seconds, while DBN's takes 102 seconds[9].

S. Ustebay [2018] The CICIDS2017 dataset, the largest publicly available dataset, is utilized to analyze a performance of a proposed system. It seeks to identify the most effective features that may meaningfully differentiate the data and assess the effects of the features on a data set after address a problems introduced by big data. As a result, the relevance value of the features is determined and recursive feature reduction is performed using a random forest. Using the collected features, the DMLP structure can identify intrusions with a 91% success rate. [10].

NID is a crucial component of any safe network. Common detection systems nowadays train an ID model from historical incursion data using historical ML techniques. An issue with these approaches is their low detection rate. A more advanced technology called data logging (DL) automatically extracts data from samples. The study's authors saw the need for a more accurate solution to NID, and so they adapted the CNN algorithm to the problem. The automated extraction of useful features from the model enables classification of intrusion samples. In tests using the KDD99 datasets, the suggested approach was found to significantly improve ID accuracy.

III. RESEARCH METHODOLOGY

Following sections provide the research methodology process for the NIDS.

A. Problem Statement

NIDSs are in high demand as a means of protecting against the ever-increasing frequency and severity of cyberattacks. Cyberattack detection and prevention is a major focus of current study. Existing NIDSs rely on antiquated ML algorithm that are both ineffective and unsuitable for the emerging, unpredictable cyber-attacks. An important barrier to evaluating network IDS performance is the lack of a comprehensive network-based data set. In the literature, most network-based approaches were tested on a NSL-KDD & UNSW-15 datasets. The methods of machine learning & deep learning will be utilized in this effort, with the NSL-KDD & UNSW-15 datasets serving as a test bed. It is widely used as a testbed for network security measures.

B. Proposed Methodology

A study is based on two distinctive datasets: UNSW & NSL-KDD, each of which has undergone particular preprocessing processes. The UNSW dataset is split into 10 distinct types of information after the null values have been removed: backdoors, shellcode, worms, reconnaissance, analysis, fuzzers, DoS, exploits, and generic. One-hot encoding is then utilized to represent a categorical variable after a data has been translated into these ten classes. Following that, the dataset is examined for class balance, which reveals a considerable class imbalance, with 196,396 samples in a majority class & 61,277 in a minority class. Similarly, a NSL-KDD dataset is preprocessed, with null values removed and classification into five classes: R2L, U2R, Probe, DoS, and Normal. Following categorization, the classes are represented using one-hot encoding. There is a class imbalance, with 130,441 samples in a majority class & 18,076 samples in a minority class. To overcome this imbalance, the combination of OSS & SMOTE is employed to decrease a majority class samples & augment a minority class. Then both data is then split into 80:20 training & testing sets. Following that, a data is scaled with a min-max scaler to ensure that features are on a consistent scale and reshape the data. The features are then extracted using Stacked CNN. The first and second CNN layers are accompanied by max-pooling layers. A goal of this design is to capture subtle patterns and relationships in data. Following feature extraction, apply the XGBoost and LightGBM models, as well as the Voting classifier, with the features obtained from the Stacked CNN as inputs. These ensemble models should improve predicted performance and produce more robust outcomes.

C. Data Collection

Both an UNSW-NB15 & the NSL-KDD databases are available for no cost, and have been heavily utilized by a number of different research projects. KDDCUP99 and NSL-KDD have both been utilized extensively in a field of ID for quite some time. [25]. The Revathi study demonstrates the usefulness of NSL-KDD datasets for evaluating various ID methods.

The 42 dimensions of each intrusion record in this dataset are broken down as follows: 38 dimensions of digital features; 3 dimensions of symbol features; and labels for traffic types. The label includes both standard information and details on four parts of attacks (DoS, R2L, Probe, and U2R). Experiments in this paper are conducted using the NSL-KDD dataset, specifically its test set (KDDTest+) & training set (KDDTrain +) for a model's training and testing phases, respectively.

Using the block diagram in Figure 1, we can see how the proposed method partitioned the whole set of connection records into two subsets, one containing 82337 test connection recordings and the other 175343 train connection recordings, and then each subset was partitioned into ten groups of ten

***Article Received:** 26 August 2023 **Revised:** 20 October 2023 **Accepted:** 02 November 2023*

records. The partitioned dataset includes a total of 42 characteristics with parallel class labels, including nine unique assaults.

D. Data Preprocessing

The data preparation phase is crucial to the success of any project. Preprocessing data prepares it for analysis and interpretation by a computer. This study uses various preprocessing which consist check null values, Encoding, one hot encoder, counting majority and minority sample, scaling and data reshaping. These processes are described below:

i. **Encoding:** The NSL-KDD dataset is divided into five groups, each of which represents a distinct type of network activity. These classes are 'dos' (45,927, 'normal' (67,343), 'r2l' (995), 'probe' (11,656), & 'u2r' (52). The UNSW dataset divided into ten separate categories that correspond to different types of cybersecurity threats. The class distribution is not uniform; whereas there are 56,000 occurrences of the 'Normal' class, only 130 occur in the 'Worms' class.

ii. **one hot encoder:** A popular technique for converting categorical information to numerical characteristics is one-hot encoding (OHE).

iii. **Counting majority and minority sample;** Samples from the majority class total 130441 in the NSL KDD dataset, with 18076 coming from the minority class. Out of a total of 196396 samples, the UNSW dataset contains 61277 members of a minority class.

iv. **Scaling:** Values can be normalized between 0 and 1 with the help of a tool called the MinMax Scaler. Its name comes from the fact that its normalization relies on the feature's highest and lowest values. [11].

v. **Data reshaping:** The data reshaping, in which the dataset is reshaped into an 11x11 matrix, is a unique component of this research.

- **One-side selection (OSS):** OSS classifies most samples into four categories: noisy, borderline, redundant, and safe. OSS finds and eliminates unnecessary samples. [12]
- **SMOTE:** The imbalanced classification issue might be solved using oversampling techniques. In order to change the empirical distribution, oversampling approaches primarily aim to increase a number of samples from a minority class. Synthetic data samples can be created using oversampling techniques, with SMOTE being one of the first and most used. On the line segments connecting the minority class's existing instances, it generates new synthetic examples. Conversely, SMOTE places equal weight on all minority samples.[13].
- **Data splitting:** In this experiment, the datasets are splitted into training & testing sets with an 80:20 ratio.
- **Feature extraction using CNN:** A CNN is an ANN typically utilized for feature extraction and classification in high-dimensional data. CNN is optimized for the reorganization of two-dimensional shapes that is highly distortion-resistant to translation, scaling, skewing, and other operations. Layers of feature extraction, mapping, and subsampling make up the architecture. It is possible to add fully connected output layers on top of a CNN's convolutional and subsampling layers[14].

E. Machine learning Classification model:

ML is a branch of AI that deals with a development of automated systems for the extraction of useful information from large datasets. To train a computer to learn and make decisions without being explicitly programmed, a method known as machine learning has been developed(Liu et al., 2020). In this study, After the features have been extracted, features are inputs in the XGBoost and LightGBM models, as well as the Voting classifier.

a) **LGBM:** It is an improved version of the "gradient boosting framework" that uses the decision tree method, and it achieves excellent results. Two of its main applications are ranking and categorization. It divides the tree in half, leaf by leaf, using the best fit method. By analyzing the variation of the statistics, which can be done in a variety of ways for better data collecting, it can be calculated(Srivastava and Dwivedi, 2022).

b) **XG boost:** Using decision trees as its weak learners, it is an enhanced version of the conventional gradient boosting method. XGBoost's popularity comes from the fact that it can offer insightful solutions to problems with structured data using a variant of the gradient-boosted trees method. In the context of gradient-boosting regression, the weak learner is represented by individual regression trees that assign a continuous score to each input data point, as shown by the corresponding leaf node. (Fatima and Pasha, 2017).

c) **Voting Classifier:** In the field of ML, a voting classifier refers to an estimator that leverages the collective knowledge of multiple autonomous prediction algorithms. The utilization of majority vote as the aggregating criterion for each estimator output is a viable option. A ML model capable of being trained on several datasets and utilized for predicting a most probable category. There exist two distinct categories of vote classifiers, namely soft & hard classifiers. (Ahamed, Arya and Nancy, 2022). The voting consists of a combination of XG boost and LGBM.

Article Received: *26 August 2023* ***Revised:*** *20 October 2023* ***Accepted:*** *02 November 2023*

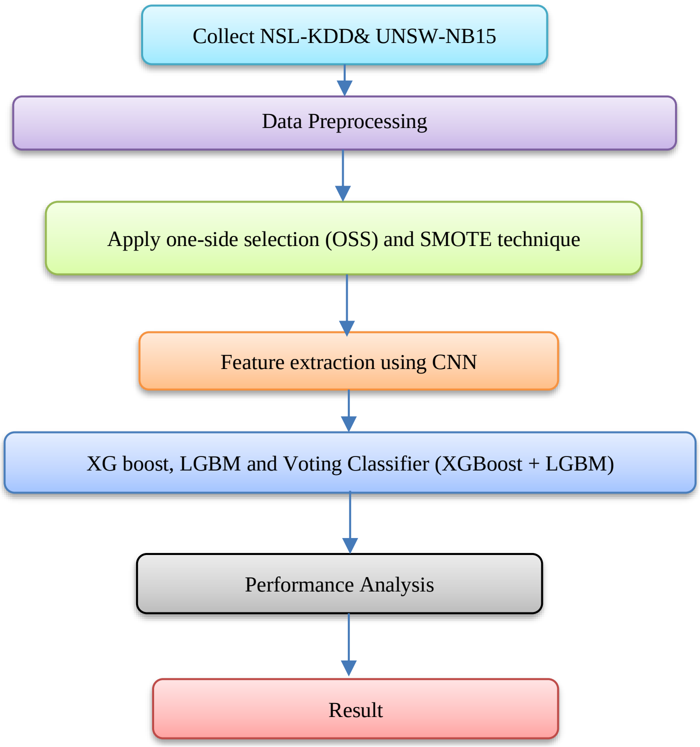

Collect NSL-KDD& UNSW-NB15
Data Preprocessing
Apply one-side selection (OSS) and SMOTE technique
Feature extraction using CNN
XG boost, LGBM and Voting Classifier (XGBoost + LGBM)
Performance Analysis
Result

Fig. 1: Block chart of Proposed Methodology

IV. RESULT ILLUSTRATION

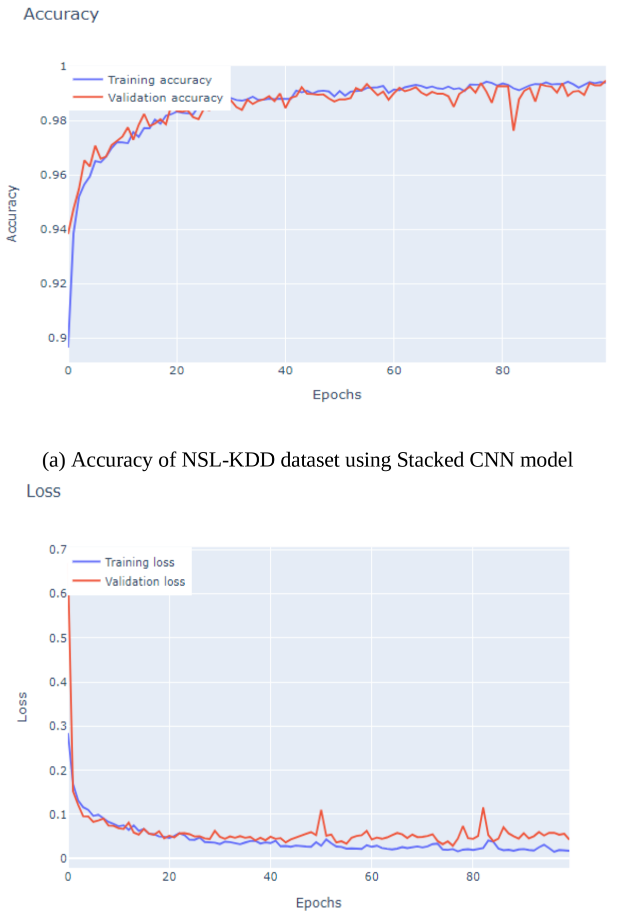
The experiments applied NumPy, pandas, seaborn, Matplotlib, TensorFlow. Network model utilized in this article had its learning rate set to 0.001. The regularization process utilized a dropout with a weight inactivation rate of 0.5, 100 iterations of the experiment, & a batch size of 128. Time spent training a detection model is cut short by eliminating unnecessary data from the majority class using stacked-CNN and XGBoost.

1) For a NSL-KDD dataset

A following section provide a NSL-KDD dataset using the proposed Stacked CNN and Voting classifier (XGBoost and LGBM). Table 1 shows the NSL-KDD dataset performance of stacked CNN model.

TABLE I
NSL-KDD PERFORMANCE

Performance	Stacked CNN model for the UNSW-15 Dataset
Training accuracy	99.58
Validation accuracy	99.47
Training Loss	0.0116
Validation Loss	0.0428



(a) Accuracy of NSL-KDD dataset using Stacked CNN model

(b) Loss of NSL-KDD using Stacked CNN model

Fig. 2. Accuracy/loss Graph of NSL-KDD using Stacked CNN model

***Article Received:** 26 August 2023 **Revised:** 20 October 2023 **Accepted:** 02 November 2023*

Figure 2 shows the Accuracy/loss Graph of NSL-KDD using stacked CNN model. Proposed model reduces validation loss 0.0428 and training loss 0.0116, respectively. While train accuracy of 99.58%, and validation accuracy of 99.47%, respectively.

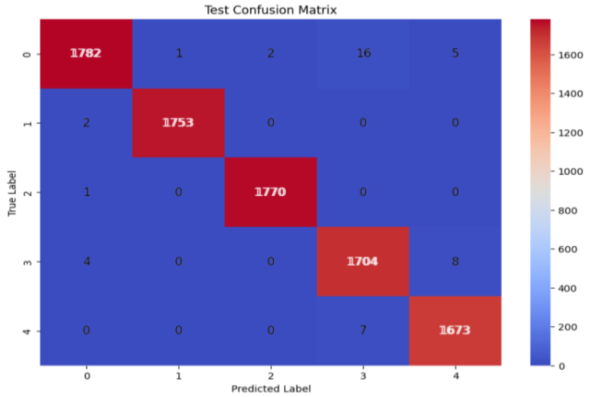


Fig. 3. Confusion Matrix of NSL-KDD using Stacked CNN model

Confusion matrix of NSL-KDD is shown in Fig. 3. Although a stacked CNN's confusion matrix can aid in the extraction of spatial attributes, it has difficulty learning sequence correlation information & sidesteps a problem of long-term information reliance. This means that NID's accuracy using only CNN alone needs to be improved.

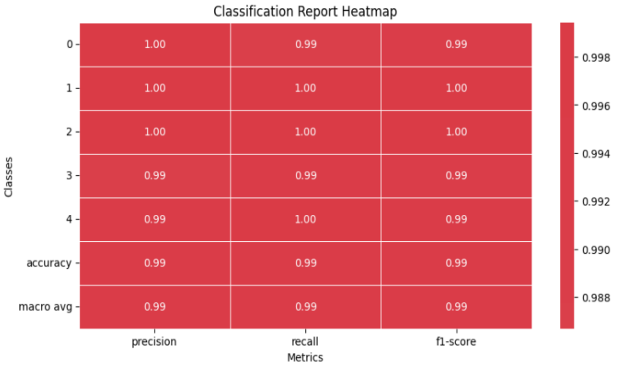


Fig. 4: Classification report heatmap of NSL-KDD using Stacked CNN model

The classification report heatmap of NSL-KDD using Stacked CNN model shows in figure 4. Proposed stacked CNN model obtain 99% performance of classification parameters like precision, recall, accuracy, and f1-score.

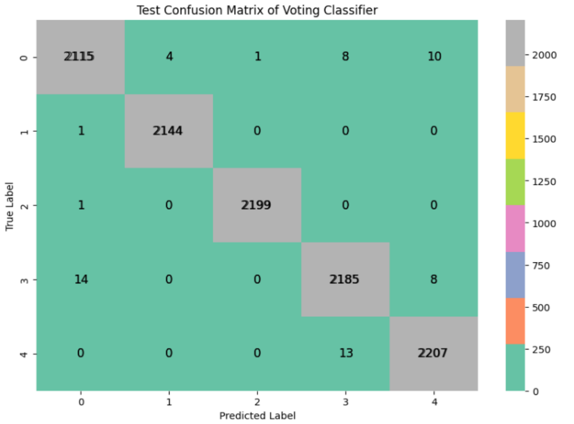


Fig. 5. Confusion Metrix of NSL-KDD using Voting classifier

The following figure 5 displays a confusion Metrix of NSL-KDD dataset using a voting classifier. In Metrix highly predicted class is 4 with number of attacks 2207, respectively.

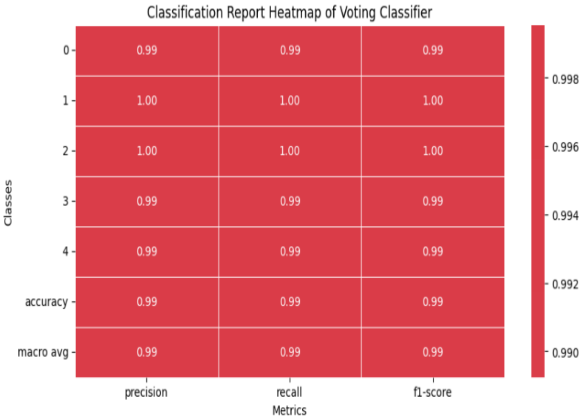


Fig. 6. Classification report heatmap of NSL-KDD using Voting classifier

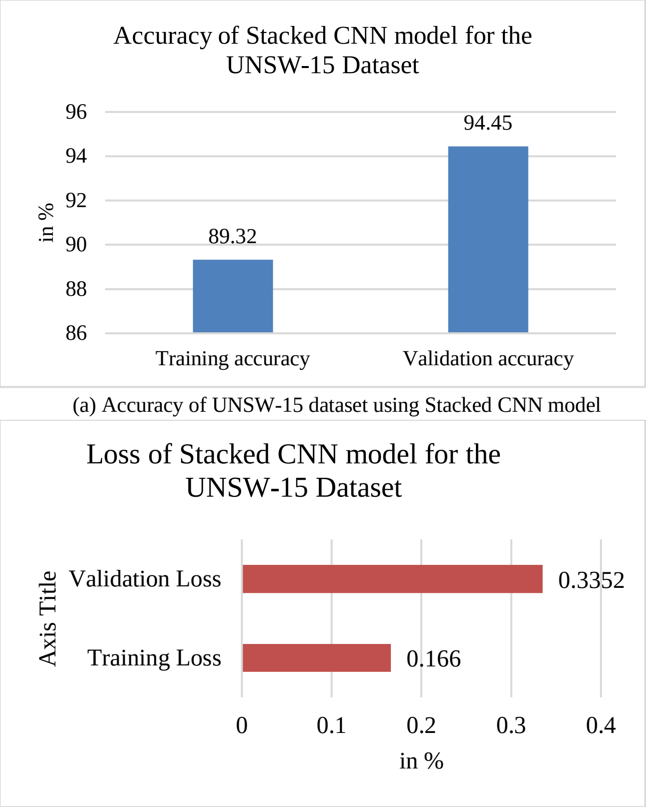
The classification report heatmap of NSL-KDD using Voting model shows in figure 6. Proposed Voting model obtain 99% performance of classification parameters like F1-score, Accuracy, Recall, and Precision.

2) For the UNSW-15 dataset

A following section provide the UNSW-15 dataset using the proposed Stacked CNN and Voting classifier (XGBoost and LGBM). Table 2 shows a NSL-KDD dataset performance of stacked CNN model.

TABLE II
UNSW-15 PERFORMANCE

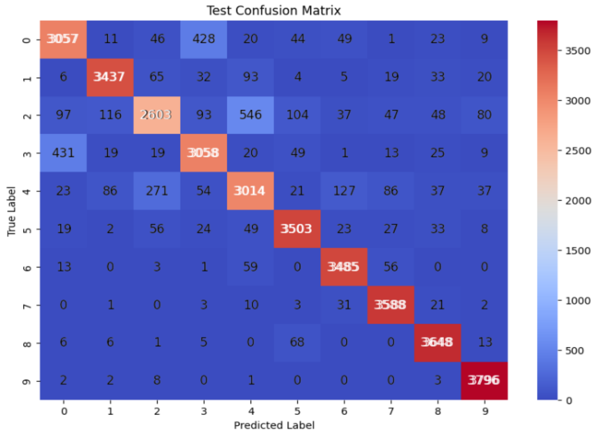
Performance	Stacked CNN model for the UNSW-15 Dataset
Training accuracy	89.32
Validation accuracy	94.45
Training Loss	0.1660
Validation Loss	0.3352



(a) Accuracy of UNSW-15 dataset using Stacked CNN model

(b) Loss of UNSW-15 using Stacked CNN model

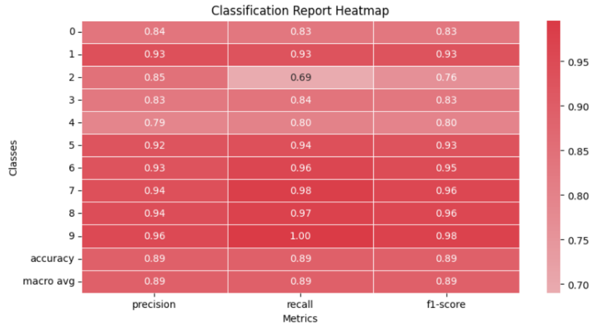
Fig. 7. Accuracy/loss Graph of UNSW-15 using Stacked CNN model. Proposed model reduces validation loss 0.3352 and training loss 0.1660, respectively. While train accuracy of 0.9445%, and validation accuracy of 0.8932%, respectively.



	0	1	2	3	4	5	6	7	8	9
0	3057	11	46	428	20	44	49	1	23	9
1	6	3437	65	32	93	4	5	19	33	20
2	97	116	2603	93	546	104	37	47	48	80
3	431	19	19	3058	20	49	1	13	25	9
4	23	86	271	54	3014	21	127	86	37	37
5	19	2	56	24	49	3503	23	27	33	8
6	13	0	3	1	59	0	3485	56	0	0
7	0	1	0	3	10	3	31	3588	21	2
8	6	6	1	5	0	68	0	0	3648	13
9	2	2	8	0	1	0	0	0	3	3796

Fig. 8. Confusion Matrix of UNSW-15 using Stacked CNN model

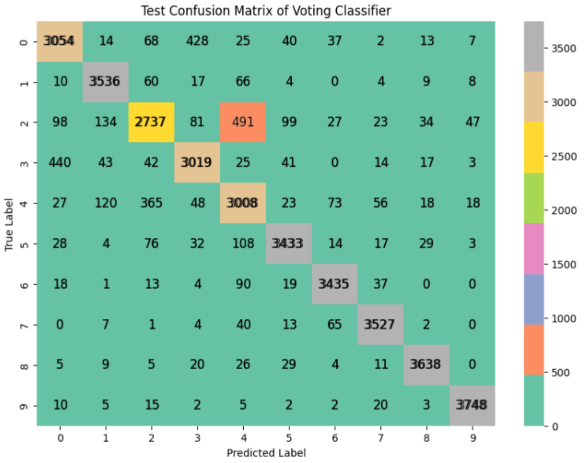
The following figure 8 shows the confusion Metrix of UNSW-15 dataset using the Stacked CNN classifier. In Metrix highly predicted class is 9 with number of attacks 3796, respectively.



Classes	precision	recall	f1-score
0	0.84	0.83	0.83
1	0.93	0.93	0.93
2	0.85	0.69	0.76
3	0.83	0.84	0.83
4	0.79	0.80	0.80
5	0.92	0.94	0.93
6	0.93	0.96	0.95
7	0.94	0.98	0.96
8	0.94	0.97	0.96
9	0.96	1.00	0.98
accuracy	0.89	0.89	0.89
macro avg	0.89	0.89	0.89

Fig. 9. Classification report heatmap of UNSW-15 using Stacked CNN model

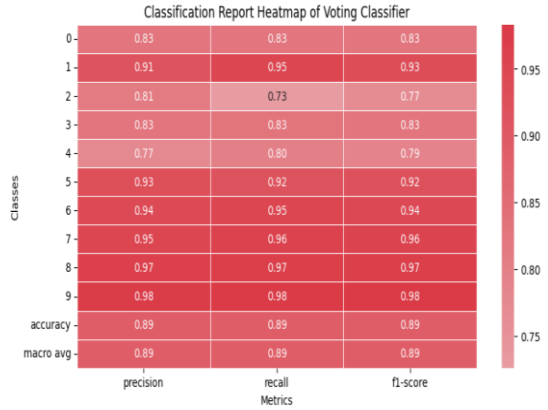
The classification report heatmap of UNSW-15 using Stacked CNN model shows in figure 9. Proposed Stacked CNN model obtains 89% performance of classification parameters like accuracy, precision, recall and f1-score.



	0	1	2	3	4	5	6	7	8	9
0	3054	14	68	428	25	40	37	2	13	7
1	10	3536	60	17	66	4	0	4	9	8
2	98	134	2737	81	491	99	27	23	34	47
3	440	43	42	3019	25	41	0	14	17	3
4	27	120	365	48	3008	23	73	56	18	18
5	28	4	76	32	108	3433	14	17	29	3
6	18	1	13	4	90	19	3435	37	0	0
7	0	7	1	4	40	13	65	3527	2	0
8	5	9	5	20	26	29	4	11	3638	0
9	10	5	15	2	5	2	2	20	3	3748

Fig. 10. Confusion Metrix of UNSW-15 using Voting classifier

The following figure 10 shows the confusion Metrix of UNSW-15 dataset using the voting classifier. In Metrix highly predicted class is 9 with number of attacks 3748, respectively.



Classes	precision	recall	f1-score
0	0.83	0.83	0.83
1	0.91	0.95	0.93
2	0.81	0.73	0.77
3	0.83	0.83	0.83
4	0.77	0.80	0.79
5	0.93	0.92	0.92
6	0.94	0.95	0.94
7	0.95	0.96	0.96
8	0.97	0.97	0.97
9	0.98	0.98	0.98
accuracy	0.89	0.89	0.89
macro avg	0.89	0.89	0.89

Fig. 11: Classification report heatmap of UNSW-15 using Voting classifier

The classification report heatmap of UNSW-15 using Voting model shows in figure 11. Proposed Voting model obtain 89% performance of classification parameters like f1-score, accuracy, precision, & recall.

V. CONCLUSION

A significant change in people's way of life has occurred as a result of the Internet's recent technological breakthroughs, which have made it an indispensable tool in daily life. Despite the fact that a number of network attack techniques are constantly being updated in the current network environment, the scope of their influence is expanding, the frequency of attacks is increasing, and the threats to network security are getting greater. In this post, we'll look at a novel IDS that makes use of hybrid sampling and DHNs and see how it works. The model was trained with data provided by SMOTE&OSS, which ensured a well-rounded training set. To some extent, it solves a problem of insufficient training by imbalanced data sets and has a potential to cut down training time for models by half. Additionally, a Network Data Preparation Method was established, which is suitable for the proposed DHN paradigm, for dealing with complex, multidimensional cyber threats. For more complex data categorization tasks, a layered network model can be constructed using stacked CNN with voting. Characteristics are automatically gathered by the model using recurrent multi-level learning, a strength of deep learning. Both a UNSW-NB15 & NSL-KDD intrusion datasets were utilized in an assessment of a suggested technique. Based on statistical significance tests, it is reasonable to infer that the suggested method excels over competing classifiers.

VI. FUTURE SCOPE

The results of the experiments show that the model can increase the effectiveness of IDS while also enhancing the accuracy of intrusion detection. In the near future, we will have access to deep learning-based classifiers that can sort data into useful categories and make our systems more accurate and economical. Improved ML and AI in IDSs are allowing for a higher detection rate. To create an online ID model, we want to modify the DLNID framework to function with a practical, integrated network capture module in the near future. A train and test data set with two goal values (one aberrant and one normal) is used in this piece. All known forms of assault are classified as abnormal traffic, while all other network activity is considered normal.

VII. ACKNOWLEDGEMENT

Funding

Not Applicable
Compliance with ethical standards

Conflict of interest

No conflicts of interest were reported by the authors.

Ethical approval

There are no studies by any of the authors in this article that used humans or animals as subjects.

Authors' contributions

Dr. Komal Prasad Chourasia oversaw the entire process and offered insightful feedback at every turn. The final manuscript was read and approved by every author.

REFERENCES

[1] A.J. Wilson and S. Giriprasad, "A Feature Selection Algorithm for Intrusion Detection System Based on New Meta-Heuristic Optimization." *J. Soft Comput Eng. Appl.*, vol. 1, no. 1, 2020.

[2] A.K. Shukla, 'Detection of anomaly intrusion utilizing self-adaptive grasshopper optimization algorithm."*Neural Comput Appl*, pp. 1–21, 2020.

[3] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm."*Appl. Soft. Comput*, vol. 8, No. 1, pp. 687–697, 2008.

[4] I. Ahmad, M. Basheri, M.J. Iqbal, and A. Rahim, "Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection." *IEEE access*, vol. 6, pp. 33789– 33795, 2018.

[5] I. Syarif, R.F. Afandi, and F.A. Saputra, "Feature Selection Algorithm for Intrusion Detection Using Cuckoo Search Algorithm." *in 2020International Electronics Symposium (IES)*, pp. 430–435, 2020.

[6] J. Ding, Q. Wang, Q. Zhang, Q. Ye, and Y. Ma, "A hybrid particle swarm optimization-cuckoo search algorithm and its engineering applications." *Math Probl Eng.*, vol. 2019.

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

[8] Kanakarajan, N.K.; Muniasamy, K. "Improving the accuracy of intrusion detection using GAR-forest with feature selection."*In Proceedings of the 4th International Conference on Frontiers in Intelligent Computing: Theory and Applications (FICTA) 2015*, Durgapur, India, 16–18 November 2015; Springer: New Delhi, India, 2016; pp. 539–547.

[9] L. Dhanabal and S. P. Shanthara, ''A study on NSL-KDD dataset for intrusion detection system based on classification algorithms.'' *Int. J. Adv. Res. Comput. Commun. Eng.*, vol. 4, no. 6, pp. 446–452, 2015.

[10] M. Diale, C. Van Der Walt, T. Celik, and A. Modupe, "Feature selection and support vector machine hyper-parameter optimisation for spam detection."*in 2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech)*, pp. 1–7, 2016.

[11] M.M.A. Zahra, M.J. Mohsin, and L.A. Abdul-Rahaim, "Artificial intelligent smart home automation with secured camera management-based GSM, cloud computing, and Arduino." *Period. Eng. Nat. Sci.*, vol. 8, no. 4, pp. 2160–2168, 2020.

[12] P. Shunmugapriya and S. Kanmani, "A hybrid algorithm using ant and bee colony optimization for feature selection and classification (AC-ABC Hybrid)." *Swarm Evol Comput*, vol. 36, pp. 27–36, 2017.Pervez, M.S.; Farid, D.M. "Feature selection and

intrusion classifi-cation in NSL-KDD Cup 99 dataset employing SVMs."*In Proceedings of the 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014)*, Dhaka, Bangladesh, pp. 1–6, 18–20 December 2014.

[13] Q.R.S. Fitni and K. Ramli, "Implementation of ensemble learning and feature selection for performance improvements in anomaly-based intrusion detection systems." *in 2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, pp. 118–124, 2020.

[14] S. Dwivedi, M. Vardhan, S. Tripathi, and A.K. Shukla, "Implementation of adaptive scheme in evolutionary technique for anomaly-based intrusion detection." *Evol Intell*, vol. 13, No. 1, pp. 103–117, 2020.

[15] S. OUIAZZANE, M. ADDOU and F. BARRAMOU, "A Multi-Agent Model for Network Intrusion Detection." *2019 1st International Conference on Smart Systems and Data Science (ICSSD)*, pp. 1-5, 2019.
Doi: 10.1109/ICSSD47982.2019.9003119.

[16] S. Revathi and A. Malathi, ''A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection.'' *Int. J. Eng. Res. Technol*., vol. 2, no. 12, pp. 1848–1853, 2013.

[17] S. Sapre, K. Islam and P. Ahmadi, "A Comprehensive Data Sampling Analysis Applied to the Classification of Rare IoT Network Intrusion Types." *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1-2, 2021.Doi: 10.1109/CCNC49032.2021.9369617.

[18] S. Sarvari, N.F.M. Sani, Z.M. Hanapi, and M.T. Abdullah, "An efficient anomaly intrusion detection method with feature selection and evolutionary neural network." *IEEE Access*, vol. 8, pp. 70651–70663, 2020.

[19] S. Sivanantham, R. Abirami and R. Gowsalya, "Comparing the Performance of Adaptive Boosted Classifiers in Anomaly based Intrusion Detection System for Networks." *2019 International Conference on Vision towards Emerging Trends in Communication and Networking (ViTECoN)*, pp. 1-5, 2019.
Doi: 10.1109/ViTECoN.2019.8899368.

[20] S. Zheng, "Network Intrusion Detection Model Based on Convolutional Neural Network." *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2021, pp. 634-637, 2020.Doi: 10.1109/IAEAC50856.2021.9390930.

[21] W.A.H.M. Ghanem, A. Jantan, S.A.A. Ghaleb, and A.B. Nasser, "An Efficient Intrusion Detection Model Based on Hybridization of Artificial Bee Colony and Dragonfly Algorithms for Training Multilayer Perceptrons." *IEEE Access*, vol. 8, pp. 130452–130475, 2020.

[22] X.C. Guo, J.H. Yang, C.G. Wu, C.Y. Wang, and Y.C. Liang, "A novel LS-SVMs hyper-parameter selection based on particle swarm optimization." *Neurocomputing*, vol. 71, no. 16–18, pp. 3211–3215, 2008.

[23] A. K. M. Mashuqur Rahman Mazumder, N. Mohammed Kamruzzaman, N. Akter, N. Arbe and M. M. Rahman, "Network Intrusion Detection Using Hybrid Machine Learning Model," 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT), Bhilai, India, 2021, pp. 1-8, doi: 10.1109/ICAECT49130.2021.9392483.

[24] Umair, Muhammad & Iqbal, Zeshan & Faraz, Muhammad & Khan, Muhammad & Zhang, Yudong & Razmjooy, Navid & Kadry, Sefedine. (2022). A Network Intrusion Detection System Using Hybrid Multilayer Deep Learning Model. Big data. 10.1089/big.2021.0268.

[25] Z. Lin and D. Hongle, "Research on SDN intrusion detection based on online ensemble learning algorithm." *2020 International Conference on Networking and Network Applications (NaNA)*, pp. 114-118, 2020.
Doi: 10.1109/NaNA51271.2020.00027.